Iterative Solvers for Modeling Mantle Convection with Strongly Varying Viscosity

Dissertation

zur Erlangung des akademischen Grades doctor rerum naturalium
(Dr. rer. nat.)

vorgelegt dem Rat der Chemisch-Geowissenschaftlichen Fakultät der
Friedrich-Schiller-Universität Jena
von Christoph Köstler[1]
geboren am 03.06.1975 in Sondershausen

[1]Staatsexamen für das Lehramt an Gymnasien in Mathematik und Physik

Gutachter:

1. Prof. Dr. Uwe Walzer, Geodynamik, Friedrich-Schiller-Universität Jena
2. Prof. Dr. Arnd Meyer, Numerische Analysis, Technische Universität Chemnitz
3. Dr. John R. Baumgardner, Computational Geodynamics, Theoretical Division, Los Alamos National Laboratory (retired)

Tag der öffentlichen Verteidigung: 13.04.2011

Abstract

This dissertation presents significant improvements to the spherical finite-element discretization and to the iterative solver of the Stokes equations within the high-performance mantle convection code Terra.

For this purpose, a stabilized Q_1-Q_1 finite-element discretization of the Stokes equations in a two-dimensional square domain has been studied in terms of evaluating its spectral properties depending on grid spacing, viscosity model and viscosity contrast. It could be shown that the spectrum of the Schur complement S becomes independent of the grid spacing when the stabilization as proposed by Dohrmann and Bochev (2004) is applied. To get this spectrum also independent of the viscosity contrast, S has to be scaled by the diagonal of the viscosity-weighted pressure mass matrix M_η. M_η is also spectral equivalent to the stabilization matrix C.

The above-mentioned finite-element discretization has been extended to a Stokes solver study tool (SSST). which has been used to compare three Krylov subspace methods: Pressure Correction, MINRES and a CG method using a block-triangular preconditioner, proposed by Bramble and Pasciak (1988). Except MINRES, all solvers have been transformed to a restarted version, using an inner-outer scheme with inner and outer stopping criteria derived from eigenvalue estimates. In the comparison, emphasis was on performance and on robustness with respect to viscosity and to iteration parameter choices. The study revealed that the difference between the Krylov solvers was less than a factor of two. However, the pressure correction algorithm showed slightly the best performance while being the simplest method to implement.

To improve the spherical finite-element discretization in Terra, the same stabilization matrix C has been included as in SSST. It is ready to use on grids with at least 84.5 millions of nodes, on coarser grids it must be weighted as the projection of the pressure to a piecewise constant function leads to a higher maximum divergence error. An adaptive weighting of C has been implemented into the solver of Terra.

From the findings of the two-dimensional study, the pressure correction algorithm of Terra has been refined and prior to solving, also S is scaled as in SSST, which leads to iteration numbers considerably less dependent on the viscosity variation than before. By applying the variable-viscosity mass matrix scaling, the total number of multigrid iterations in the first ten time steps of a convection simulation could be reduced by a factor

of four in the presence of strong lateral viscosity variations. This improvement could be even larger, up to a factor of 20, if the convergence of the multigrid solver, which is used for calculating velocities and velocity search directions, would not depend that much on the viscosity variations. Volume-weighted harmonic viscosity averaging has been introduced to Terra to apply cellwise constant viscosities in M_η and C. These improvements allow to model the convection of Earth's mantle more realistically.

Zusammenfassung

Diese Dissertation beschreibt wesentliche Verbesserungen in der Finite-Elemente Diskretisierung sowie in dem iterativen Lösungsverfahren für die Stokes-Gleichungen des sphärischen Mantelkonvektionsprogramms Terra. Besondere Berücksichtigung fand dabei die effiziente Ausnutzung von Höchstleistungsrechnern.

Dazu wurde zunächst in einem zweidimensionalen quadratischen Gitter eine stabilisierte Finite-Elemente Diskretisierung mit jeweils stückweise linearen Ansatzfunktionen für Druck und Geschwindigkeit implementiert. Die Stabilisierung folgt dem Vorschlag von Dohrmann and Bochev (2004) und verwendet eine Projektion des Drucks auf eine stückweise konstante Funktion. Unter Variation der Gitterfeinheit, des Viskositätsmodells und des Viskositätskontrasts wurde die stabilisierte Diskretisierung auf ihre spektralen Eigenschaften hin untersucht. Die Stabilisierung bewirkt dabei eine Gitterunabhängigkeit des Spektrums des Schurkomplements S. Die Unabhängigkeit vom Viskositätsmodell sowie vom Viskositätskontrast wird durch Präkonditionierung von S mit einer viskositätsgewichteten Massenmatrix M_η oder durch Skalierung mit deren Diagonale erreicht. M_η ist spektral äquivalent zur Stabilisierungsmatrix C.

Diese Diskretisierung wurde zu einem Studienwerkzeug für Stokes-Löser (SSST) weiterentwickelt, um damit verschiedene Löser hinsichtlich ihrer Robustheit gegenüber Viskositätsvariationen und evtl. nicht optimal gewählten Abbruchkriterien zu vergleichen. Es wurden drei Krylov-Unterraumverfahren untersucht: Druckkorrektur-Verfahren (PC), Verfahren der minimierten Residuen (MINRES) und ein konjugiertes Gradientenverfahren mit einem speziellen, von Bramble and Pasciak (1988) entwickelten Blockpräkonditionierer (BPCG). Bis auf MINRES wurden die Löser in einer äußeren Schleife mit passend gewählten Abbruchkriterien mehrfach gestartet. Sämtliche Abbruchkriterien wurden aufgrund von Eigenwertabschätzungen berechnet. Dabei stellte sich heraus, dass die Unterschiede zwischen den Lösern gering waren, in den meisten Fällen weniger als Faktor 2 in der Rechenzeit. Das Druckkorrektur-Verfahren ist geringfügig schneller als die beiden anderen Löser bei starken Viskositätskontrasten und ist am leichtesten zu implementieren.

Die Stabilisierung wurde nach der gleichen Methode wie in SSST in Terra implementiert und kann ohne Einschränkung auf Gittern mit mindestens 85,4 Millionen Knoten verwendet werden. Auf gröberen Gittern

muss C gewichtet werden. Eine adaptive Wichtung wurde in den Terra-Löser implementiert.

Basierend auf den Ergebnissen der Löserstudie in SSST wurde das Druckkorrektur-Verfahren in Terra angepasst und ebenfalls in einer äußeren Schleife bei Bedarf mehrfach gestartet. Eine wesentliche Verbesserung in der Konvergenzgeschwindigkeit des Stokes-Lösers brachte die Anwendung der viskositätsgewichteten Massenmatrix M_η bzw. deren Diagonale zur Skalierung des Schurkomplements. Allein die Skalierung bewirkt eine Reduktion der Summe der Multigrid-Iterationen in den ersten 10 Zeitschritten einer Modellierung um einen Faktor von 4 bei dem Vorhandensein starker lateraler Viskositätsvariationen. Bei Verwendung eines Multigrid-Lösers für die Invertierung des Impulsoperators, dessen Konvergenzraten unabhängig von der Viskositätsvariation wären, würde die Iterationszahl um den Faktor 20 sinken. Um eine zellweise konstante Viskostität in dem Aufbau der Operatoren M_η und C zu verwenden, wurde eine volumengewichtete harmonische Mittelung derselben in Terra implementiert. Diese Verbesserungen sind ein wesentlicher Schritt um die Realitätsnähe von Modellen des Erdmantels weiter zu verbessern.

Acknowledgements

I would like to thank all the colleagues of the Geodynamics Group Jena. In particular I thank Uwe Walzer, its leader, for raising the funds to support this research and for providing an exceptional working atmosphere. I highly acknowledge Markus Müller for many stimulating discussions, for developing a suite of testing frameworks, for setting up the computing facilities, were the 2-D calculations of this work were executed, and for introducing the Ruby language to me. I also thank him for promoting techniques like pair programming and test driven development in our group and for proofreading. I am also grateful to Roland Hendel for his calm and peaceful influence, for the many refreshments he provided throughout my thesis work and for providing IDL routines. Further thanks are given to Andreas Hoffmann for occasional, but very effective technical support.

I would like to express my gratitude to John Baumgardner for providing the Terra code, for the encouragement and for the good advice he gave to my work. I further thank the whole Terra Group, in particular Rhodri Davies for bringing us together in Munich and in Cardiff to start a fruitful collaboration, and Marcus Mohr for the technical improvements he provided to the Terra code.

I also want to acknowledge the administrators of the HLRB2 supercomputer at LRZ Garching where the 3-D calculations of this work were executed.

I want further to thank Arnd Meyer for co-organizing the Chemnitz FEM Symposia and for sharing his experience in setting up efficient Stokes solvers. I thank all reviewers and Markus Müller for proofreading the 2-D chapters of the manuscript.

I would like to acknowledge my wife Lydia for her support and patience and for the sacrifices she made to allow this work to be finished in time. I also thank our little daughter Salome for providing happiness around her and for sleeping through the night since she was only a few weeks old. My parents and parents-in-law are also acknowledged for their support.

Most of all, I thank God, who put me in a fellowship to ponder His great works and who enabled me and granted me to finish this work.

Great are the works of the LORD; they are pondered by all who delight in them.
Psalm 111, 2

Contents

Abstract iii

Zusammenfassung v

Acknowledgements vii

1 Introduction 1
- 1.1 Convection of the Earth's Mantle . . . 1
- 1.2 Governing Equations . . . 3
- 1.3 Numerical Modeling of Mantle Convection . . . 4
  - 1.3.1 Variable Viscosity in Numerical Models . . . 5
  - 1.3.2 Modeling Convection and Evolution of the Earth's Mantle Using Terra . . . 6

2 Discretization of the Stokes Equations in 2D 9
- 2.1 Reference Problems . . . 10
- 2.2 Weak Formulation . . . 10
- 2.3 Uniqueness of the Weak Solution . . . 12
- 2.4 Velocity and Pressure Discretization . . . 14
- 2.5 Stabilization Using Local Pressure Projections . . . 15
- 2.6 Discretization Errors . . . 17
- 2.7 Spectral Properties of the Stokes Matrix . . . 17
- 2.8 Variable Viscosity . . . 18

3 Solution of the Stokes Equations in 2D 21
- 3.1 Scaling . . . 22
- 3.2 Pressure Correction Algorithm . . . 23
  - 3.2.1 Error Propagation and Solver Restart . . . 24
  - 3.2.2 Inner Solver . . . 26
  - 3.2.3 Stopping Criteria and Error Norms . . . 26
  - 3.2.4 Convergence Properties . . . 28
- 3.3 Preconditioned MINRES Algorithm . . . 29
  - 3.3.1 Preconditioning and Convergence Properties . . . 30
- 3.4 Bramble-Pasciak-CG . . . 31
  - 3.4.1 Preconditioner for A and Convergence Properties . . . 33
- 3.5 Results . . . 36

3.5.1 Inner Accuracy 37
3.5.2 Viscosity Variations 38
3.5.3 Convergence Properties and Residual Reduction . . 43
3.6 Discussion and Bibliographical Notes 43
3.7 Conclusion 47

4 3D-spherical Discretization 49
4.1 Computational Grid 49
4.2 Finite-element Operators in the Spherical Shell 51
4.2.1 Discretization of Mass and Stabilization Matrices . 52
4.2.2 Properties of the Stabilization Matrix 54
4.2.3 Weighting of the Stabilization Matrix 55
4.2.4 Effect of Stabilization to the Discretization 56
4.3 Variable Viscosity 56
4.3.1 Viscosity Averaging in the Operators 57
4.4 Time Discretization 59
4.5 Remarks on the Discretization 59

5 3D-spherical Stokes Solver 61
5.1 Example Problems 61
5.2 Scaling and Preconditioning 64
5.3 Pressure Correction Algorithm 65
5.4 Stopping Criteria and Solver Restart 65
5.5 Time Stepping 67
5.6 Convergence Results 68
5.7 Discussion and Bibliographical Notes 72

6 Summary and Conclusions 77

A Further Information to SSST 81
A.1 Detailed Results for Example 2 81

B Further Information to Terra 87
B.1 Input Parameters in the Code 87
B.2 Parallelization 88

Bibliography 91

Chapter 1

Introduction

1.1 Convection of the Earth's Mantle

Thermal convection in the Earth's mantle is in many ways connected to the life existing on its surface. As an effective way of heat transport, it plays a significant role in controlling the heat budget of our planet. It is also highly connected to plate tectonics, where for several decades an explanation for the movement of tectonic plates on the Earth as stated by Wegener (1915) had been sought. Despite quantitative convection models, given by Holmes (1931), Hales (1936) and Pekeris (1935), which confirmed convection to be a viable mechanism to transport heat through the mantle and to maintain gravity anomalies, one main objection to the idea of a convective mantle was the lack of insight how solid rock should move continuously without being molten. In the 1950s, laboratory experiments confirmed that crystalline rock is capable of slow high-viscous creep flow even though temperature is only a small fraction of the melting temperature. This, together with the observations of magnetic polar wander (Runcorn, 1956), of seafloor spreading (Hess, 1962) and of reversals of the oceanic crust's magnetization (Vine and Matthews, 1963), led to the general acceptance of the idea of thermal convection in the mantle. These historic milestones in understanding mantle convection are described in more detail by Schubert et al. (2001) and Bercovici (2007). Mantle convection and plate tectonics, yielding a high surface heat flow in some regions of the Earth, are also necessary to maintain a sufficient temperature gradient in the outer core so that convection of the iron-rich outer core provides the magnetic field which keeps us save from cosmic radiation. Whether or not plate tectonics, subduction and volcanism play a significant role in controlling the surface temperature on Earth for water to be liquid is still an open question. Liquid water is a necessary condition for life on Earth, and it is assumed also to be necessary for subduction to occur. Furthermore, subduction and volcanism play a significant role in the global carbon cycle on long timescales (Bercovici, 2007).

Current challenges in understanding mantle convection are given by

the complex variable rheology, combined with solid-solid phase transitions. The most significant of these occur at 410 km and 660 km depth as well as close to the core-mantle boundary. These phase transitions have been detected by jumps in seismic velocities and density and have been confirmed by high-pressure mineralogy (see Figure 1.1). While seismic

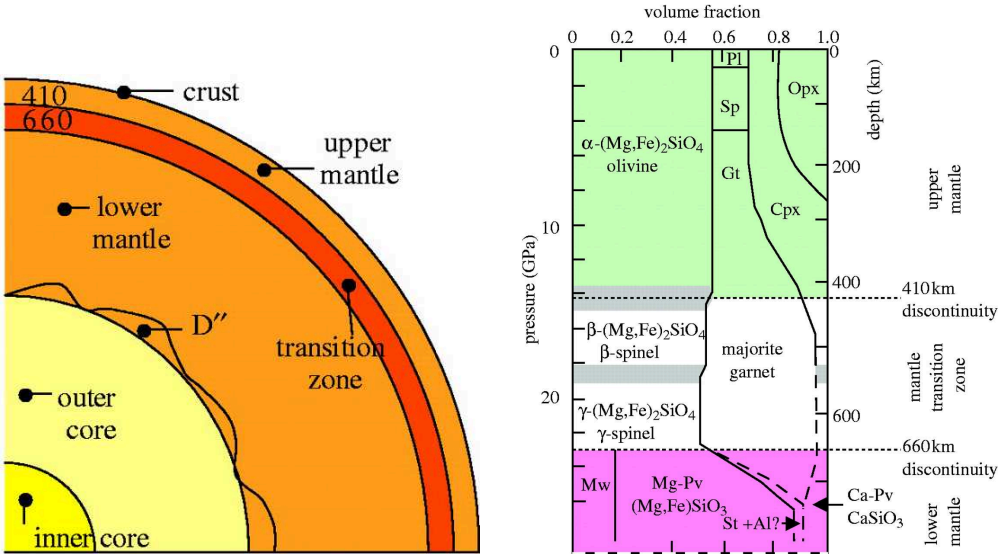

Figure 1.1: Radial structure and mineral composition of the Earth, taken from Bovolo (2005)

tomography showed that some slabs penetrate the 660 km transition (van der Hilst et al., 1997) and thus favor whole-mantle convection, mantle geochemistry observations of incompatible element abundances in mid-ocean ridge basalts (MORB) and in ocean island basalts (OIB), suggest the existence of distinct geochemical reservoirs (Hofmann, 2003). There is still no consensus whether a kind of undulating layering in the lower mantle, a thin layer at the base of the mantle or a "marble-cake" or "plum-pudding" mantle causes the heterogeneity to persist to the present time (Tackley, 2007). Another open question regards the origination of plate tectonics. How does a subduction zone initiate from a stiff lithosphere? Several assumptions for weakening and shear localization have been applied to numerical models, including visco-plastic yielding which is considered to play an important role in generating plates (Tackley, 2000a,b; Bercovici, 2003; Walzer et al., 2004b). However, these models are still very simple compared to the complex rheology of a subduction zone. The interplay of plate tectonics, chemical differentiation and continental growth through Earth's history did not receive much attention up to now and has been studied only by Walzer and Hendel (2009, 2011). Recently, also the role of water and volatiles is included in mantle convection studies (Richard and Bercovici, 2009). How are these rheological and compositional "features" represented in a convection model? In many models they lead to

strong viscosity variations with steep gradients and thus provide significant challenges within numerical models as will be seen throughout this dissertation. When rheology is absorbed in an effective viscosity, this varies by several orders of magnitude in the Earth's mantle, depending on temperature, pressure, abundances of volatiles, grain size and phase transformations. Near plate boundaries, the asthenospheric viscosity is assumed to be about 10^{18} Pa s and even lower within small regions of partial melting (see also (Billen, 2008)). Within the lithosphere as well as in the mid-lower mantle, values as high as 10^{25} Pa s are to be assumed (Walzer et al., 2004b).

1.2 Governing Equations

Usually, the treatment of convection in planetary mantles is limited to infinite Prandtl and Ekman numbers, i. e. inertial and Coriolis forces are neglected. The mathematical formulation of mantle convection consists of a Stokes equation system, comprising the conservation equations for momentum and mass:

$$-\nabla \cdot \tau + \nabla p = \rho \vec{g} \tag{1.1}$$

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \vec{u}) = 0 \tag{1.2}$$

where $\vec{u}$ is the velocity, p pressure, ρ density, $\vec{g}$ gravity and τ the shear stress tensor, and an energy equation

$$\rho c_p \frac{dT}{dt} = \nabla \cdot (k \nabla T) + Q + \alpha T \frac{dp}{dt} + \tau_{ik} \dot{\epsilon}_{ik} + p \nabla \cdot \vec{u} \tag{1.3}$$

where c_p is the specific heat at constant pressure, α the coefficient of thermal expansion, k the thermal conductivity, Q the internal heat generation rate per unit volume, $\dot{\epsilon}_{ik}$ the strain rate tensor and T the absolute temperature. In the momentum equation (1.1), $\vec{u}$ is included implicitly through τ and an appropriate formulation of rheology. In case of linear rheology, the stress tensor is related to strain rate by

$$\tau_{lm} = 2\eta \left(\dot{\varepsilon}_{lm} - \frac{1}{3} \delta_{lm} \dot{\varepsilon}_{kk} \right) \tag{1.4}$$

$$= \eta \left(\frac{\partial u_l}{\partial x_m} + \frac{\partial u_m}{\partial x_l} - \frac{2}{3} \delta_{lm} \frac{\partial u_k}{\partial x_k} \right), \tag{1.5}$$

where η is the dynamic shear viscosity. The conservation equations are supplemented by an equation of state, relating density to temperature and pressure by, e.g.,

$$\rho = \rho_r \left[1 - \alpha (T - T_r) + K_T^{-1} (P - P_r) + \sum_{k=1}^{2} \Gamma_k \Delta \rho_k / \rho_r \right] \tag{1.6}$$

where the index r refers to the adiabatic reference state, K_T is the isothermal bulk modulus, $\Delta\rho_k/\rho_r$ denotes the non-dimensional density jump for the kth phase transition and Γ_k is a measure of the relative fraction of the heavier phase. A detailed description of these equations as well as a derivation of an alternative expression of the energy equation are given in Walzer et al. (2004a).

In case of incompressible flow (relevant in Chapter 2 and in some examples in Chapter 5), the third summand in (1.5) vanishes, leading to:

$$\tau_{lm} = \eta \left(\frac{\partial u_l}{\partial x_m} + \frac{\partial u_m}{\partial x_l} \right). \tag{1.7}$$

Then, with $\rho = const$, the mass equation (1.2) simplifies to:

$$\nabla \cdot u = 0 \tag{1.8}$$

Two basic types of boundary conditions can be distinguished. Dirichlet conditions prescribe velocity and/or pressure values on the boundary, whereas Neumann conditions prescribe velocity derivatives. It is, however, possible to mix these types of boundary conditions which is also done in the Terra code. As an example, a common choice of the velocity conditions on the shell boundaries is to require a Dirichlet condition for the radial component (no in/out-flow) and Neumann conditions for the tangential components (zero gradient, i.e. no shear stress at the boundary).

1.3 Numerical Modeling of Mantle Convection

Following models of high-viscous thermal convection in a 2-D Cartesian domain (Mckenzie et al., 1974; Christensen, 1984b) to understand the mechanisms and scaling laws of heat transport within the Earth's mantle, in the 1980s the first 3-D spherical models were developed. Nevertheless, 2-D models are still in use and a field of active development in order to examine convection phenomena individually rather than aiming at an integrated Earth model. These include, among others, studies of various rheologies (Christensen, 1984a; Moresi and Solomatov, 1995; Yang and Baumgardner, 2000; Solomatov and Reese, 2008; Gerya, 2010), one-sided subduction models (Sobolev and Babeyko, 2005; Gerya et al., 2008) and regional subduction models (Gerya et al., 2006; Billen, 2008).

In three-dimensional applications, several discretization methods are commonly used to study mantle convection, these are finite-difference (FD), finite-element (FE), finite-volume (FV) and spectral methods. As parallel computing is of crucial importance here, spectral methods are less suited to gain high efficiency because of their global basis functions, but the other three methods all remain popular in this field. FD methods (Tackley, 2008) have the advantage of being very memory-efficient, while FV methods (Stemmer et al., 2006) fulfill the mass conservation exactly

and FE methods (Burstedde et al., 2009) provide the highest flexibility regarding the underlying mesh.

While constructing grids for Cartesian models, which are also widely used in mantle convection studies, is fairly straightforward, spherical models (apart from spectral methods) require a distinct starting point to set up a grid. Harder and Hansen (2005) and Stemmer et al. (2006) projected a cube onto the sphere. Zhong et al. (2000) used 12 spherical bricks to compose the shell of, which are further subdivided. These spatial discretizations are essentially based on Cartesian grids. Another approach starts from a latitude-longitude grid. As this has coordinate singularities at the poles, Kageyama and Sato (2004) put two low-latitude patches of such a grid together to obtain a so-called Yin-Yang grid. This was also applied by Tackley (2008), to extend a 3D-Cartesian code to a spherical one. As Baumgardner (1983) developed Terra from scratch, without an existing Cartesian code, he projected the regular icosahedron onto the 2-sphere as this is the Platonic body closest to the sphere. At that time, such a grid had already been successfully used in geomagnetics and oceanography, where it is still widely used today. Meanwhile, is also applied in planetary seismology (Knapmeyer, 2008) and by the German Meteorological Service (Randall et al., 2002).

In solving the discretized mass and momentum equations, almost all 3-D models use a multigrid solver for the velocity subsystem. When obtaining textbook-efficiency, it needs only $\mathcal{O}(n)$ iterations, where n is the number of unknowns. The choices of the pressure solver vary, depending also on the discretization, with multigrid (FD), Krylov subspace methods (FE) and SIMPLER algorithms (FV) being very common. This dissertation presents a study of different Krylov methods and their preconditioners and iteration parameters to FE models and an inf-sup-stabilization of the spherical-shell FE-model Terra together with a significant improvement of the variable-viscosity Stokes solver.

1.3.1 Variable Viscosity in Numerical Models

Modeling convection in the Earth's mantle with the viscosity variations mentioned in Section 1.1 has been a long-standing problem. The first studies including varying viscosity were done in two dimensions (Christensen, 1984a; Moresi and Solomatov, 1995; Moresi et al., 1996; Yang and Baumgardner, 2000). While Christensen (1984a) used a spline interpolation on a rectangular grid, the other references use finite elements. Three-dimensional variable-viscosity simulations were run by Christensen and Harder (1991), Tackley (1993, 2008), Bunge et al. (1997), Yang and Baumgardner (2000), Stemmer et al. (2006), Zhong et al. (2008) and others. One result of these simulations is that strongly temperature-dependent viscosity alone leads to a stagnant lid on top of the convective region if the viscosity contrast exceeds 4-5 orders of magnitude. Most of the temperature difference then occurs in that lid. The transition between

mobile, sluggish and stagnant-lid convection has been modeled by Moresi and Solomatov (1995); Richards et al. (2001) and Loddoch et al. (2006). Pressure-dependence in some way counteracts the build-up of a stagnant lid, it promotes longer wavelengths of the flow (Bunge et al., 1997; Tackley, 1996), sheet-like downwellings and, together with a viscoplastic yield stress, it leads to plate-tectonic behavior (Walzer et al., 2004b). Applying power-law rheology has almost the same effect as if the pressure- and temperature-dependence of viscosity would be decreased by a factor of 2 to 3 in the exponent (Christensen, 1984a).

1.3.2 Modeling Convection and Evolution of the Earth's Mantle Using Terra

One of the earliest three-dimensional numerical model was the spherical-shell model Terra, developed by Baumgardner (1983, 1985). It uses a finite-element discretization on a triangular grid, and it utilizes an efficient multigrid algorithm to solve for the velocity. It was parallelized by Bunge and Baumgardner (1995) through message passing and domain decomposition in two of three dimensions. A first study of convection with Earth-like Rayleigh number of 10^8 and depth-dependent viscosity was done by Bunge et al. (1997). At the same time, Yang (1997) improved the multigrid algorithm with matrix-dependent transfer operators to represent varying coefficients properly on coarse grids. With this code, Richards et al. (2001) investigated surface mobility as a function of viscosity variation and yield stress, and Reese et al. (2005) explored a parameter range of $\Delta\eta$ between 10^5 and 5×10^7 with an internal Rayleigh number of 10^6. Walzer et al. (2004b) derived a viscosity profile, with three high-viscosity and three low-viscosity zones and steep gradients, based on post-glacial rebound, mantle mineralogy, seismic tomography, thermodynamics and high-pressure geophysics. However, some restrictions were put into this viscosity model because of numerical reasons, especially regarding lateral variations due to temperature-dependence. Their model derived the evolution of self-consistent oceanic plates in connection with the thermal evolution of the spherical shell of the Earth. In this model it was necessary to assume that the oceanic lithosphere is also a *chemical* boundary layer and that its existence is not determined by the temperature dependence of viscosity alone.

Walzer and Hendel (2008, 2009) again showed the importance of these viscosity variations for plate tectonics and surface mobility on Earth and incorporated chemical differentiation of continents and, as a complement, of the depleted MORB mantle (DMM). In their model, continents evolve by the interplay of chemical differentiation and convection/mixing, without the requirement of modified boundary conditions on the outer surface of the shell. DMM is partly stirred into the other mantle reservoirs, resulting in a marble-cake mantle with a high concentration of DMM in the asthenosphere (Walzer and Hendel, 2011).

As usual, in spherical models the mantle is modeled as a non-rotating spherical shell, although the deviation of Earth's surface from the mean radius ranges from +7 to -14 km and that of Mars' surface from +7 to -13 km. This, however, should not significantly affect the convection and heat transport mechanisms.

Terra offers many options to include variable viscosity, phase transitions, time-dependent heating, transport of chemical species, which, in turn, also can affect density and radiogenic heating. Various boundary conditions for the lower and upper boundary of the Earth's mantle can be specified. Moreover, an approximation to compressibility is included to run the model with physical quantities of the real Earth or any other terrestrial planet.

Chapter 2

Discretization of the Stokes Equations in 2D

This chapter is devoted to the description of the Stokes solver study tool (SSST), a finite-element discretization of the Stokes equations based on the multigrid test framework of Müller (2008). SSST is an extension of this framework from a Laplace equation to a Stokes system and has been developed partly in cooperation with Markus Müller, who contributed an algebraic formulation of the stabilization matrix in MuPAD. SSST is written in GNU Octave and uses the computer algebra system MuPAD, which is now a part of Symbolic Math Toolbox in MATLAB. MuPAD functions are used for symbolic differentiation and integration in the calculation of the finite element matrices, as well as for analytical computation of the norms of the exact solutions in Section 2.1. Although SSST is able to assemble the Galerkin system for arbitrary quadrilateral grids, the examples in this chapter are limited to a square grid with quadratic elements. Here, the focus is on handling the algebraic irregularities introduced by the strongly varying viscosity coefficient. Therefore, the formulation is also limited to incompressible flow, and Eqs. (1.1), with τ given by (1.7), and (1.8) are to be solved. The well-posedness of the continuous variable-viscosity Stokes system has been demonstrated by Tabata (2006) and by Olshanskii and Reusken (2006).

The Stokes equations are discretized with a stabilized Q_1–Q_1 finite-element pair, which implies piecewise bilinear velocity and pressure functions. A stabilization block, based on local pressure projections (Dohrmann and Bochev, 2004) is added to the Galerkin system. I show numerically how this ensures the well-posedness of the discrete system. Discretization errors are computed on coarse grids. The generalized inf-sup constant δ is estimated by computation of eigenvalues of the Schur complement, scaled by the pressure mass matrix, on coarse grids. A viscosity scaled mass matrix, proposed by Olshanskii and Reusken (2006), is utilized, which causes δ to be almost independent of the viscosity variations. In the example problems considered here, the velocity is prescribed everywhere on the

boundary.

2.1 Reference Problems

In order to evaluate the accuracy of the discretization and iterative solutions, analytical examples for constant viscosity are utilized. Their extension to variable viscosity is described in Section 2.8. The first one is the Example 5.1.4 of Elman et al. (2005). It is a model of colliding flow and applies the following analytical solution to the square domain $\Omega = [-1, 1]^2$, see Figure 2.1:

$$u_x = 20xy^3, \quad u_y = 5x^4 - 5y^4, \quad p = 60x^2y - 20y^3 \tag{2.1}$$

Note, that no forcing term is needed.

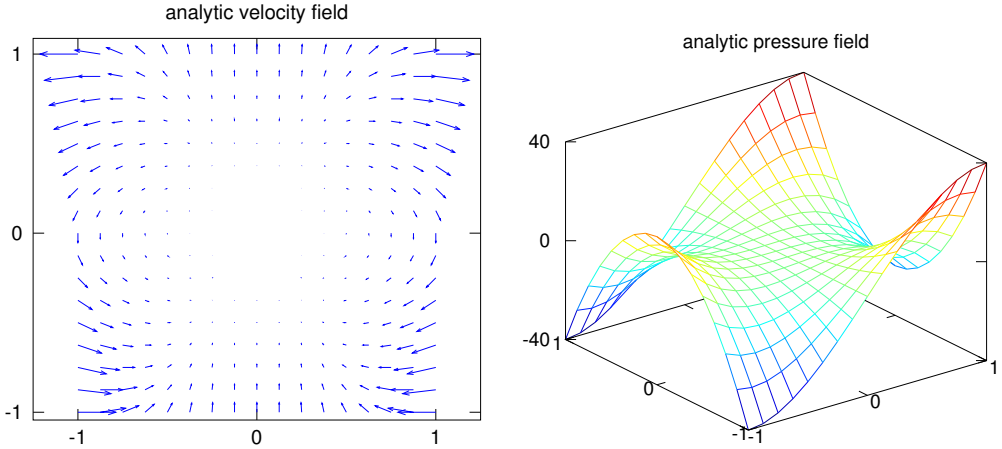

Figure 2.1: Velocity plot and pressure plot for Example 1

The second example is taken from Dohrmann and Bochev (2004) and applies the following analytical solution to the square domain $\Omega = [0, 1]^2$, see Figure 2.2:

$$\begin{aligned} u_x &= x + x^2 - 2xy + x^3 - 3xy^2 + x^2y \\ u_y &= -y - 2xy + y^2 - 3x^2y + y^3 - xy^2 \\ p &= xy + x + y + x^3y^2 - 4/3 \end{aligned} \tag{2.2}$$

The constant in p is chosen to yield $\int_\Omega p(x, y)d\Omega = 0$ and the rhs-term f is calculated from applying the Stokes equations to u and p.

2.2 Weak Formulation

A solution $(\vec{u}, p)$ to (1.1) and (1.8) is known as a classical solution. In this case $\vec{u}$ must have continuous second order derivatives and must be

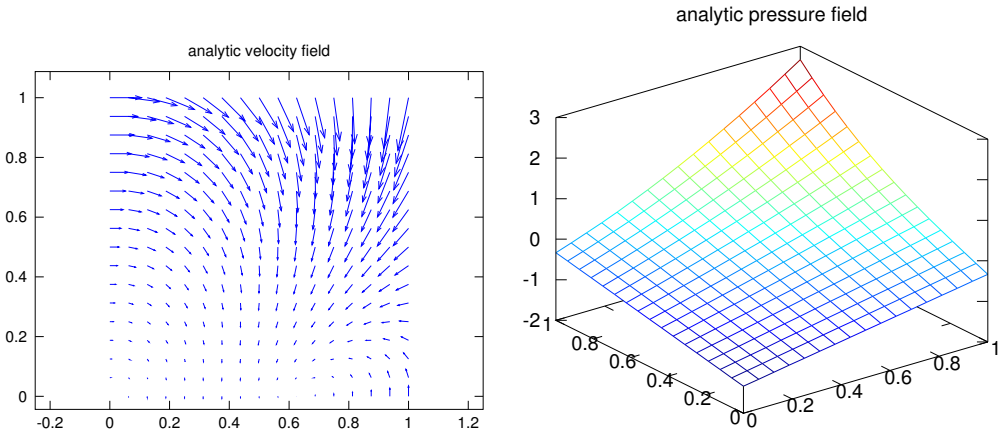

Figure 2.2: Velocity plot and pressure plot for Example 2

continuous up to the boundary. However, if viscosity or density are discontinuous within the domain, no classical solution may exist, whereas the physical problem still has a solution. To alleviate the requirements for $\vec{u}$, and to enlarge the solution space, we consider the weak integral form of the Stokes problem:

$$\int_\Omega \vec{v} \cdot (-\nabla \cdot \tau + \nabla p - \rho \vec{g}) = 0 \qquad \forall \vec{v} \in V_0 \tag{2.3}$$

$$\int_\Omega q \, \nabla \cdot \vec{u} = 0 \qquad \forall q \in L_2(\Omega) \tag{2.4}$$

with

$$V_0 = \left\{ \vec{v} \in \mathcal{H}^1(\Omega)^d \mid \vec{v} = 0 \text{ on } \partial\Omega_D \right\} \tag{2.5}$$

$$\mathcal{H}^1(\Omega) = \left\{ u : \Omega \to \mathbb{R} \;\middle|\; u, \frac{\partial u}{\partial x}, \frac{\partial u}{\partial y} \in L_2(\Omega) \right\} \tag{2.6}$$

$\mathcal{H}^1(\Omega)$ is the Sobolev space defining the continuity requirements of the velocity test functions $\vec{v}$. Therefore, we can apply integration by parts and the divergence theorem to obtain:

$$\begin{aligned} \int_\Omega -\vec{v} \cdot \nabla \cdot \tau &= \int_\Omega \nabla \vec{v} : \tau - \int_\Omega \nabla \cdot (\tau \cdot \vec{v}) \\ &= \int_\Omega \nabla \vec{v} : \tau - \int_{\partial\Omega} (\tau \cdot \vec{v}) \cdot \vec{n} \\ \int_\Omega \vec{v} \cdot \nabla p &= -\int_\Omega p \, \nabla \cdot \vec{v} + \int_\Omega \nabla \cdot (p\vec{v}) \\ &= -\int_\Omega p \, \nabla \cdot \vec{v} + \int_{\partial\Omega} p\vec{n} \cdot \vec{v} \end{aligned}$$

As the examples considered in this chapter have Dirichlet conditions over the entire boundary, the surface integrals vanish with the test functions

also vanishing on the boundary (see (2.5)). Eq. (2.3) can then be rewritten as:

$$\int_\Omega \nabla\vec{v} : \tau - \int_\Omega p\,\nabla\cdot\vec{v} = \int_\Omega \vec{v}\cdot\rho\vec{g} \tag{2.7}$$

The velocity solution itself is chosen from the space

$$V_D = \left\{\vec{u}\in\mathcal{H}^1(\Omega)^d \mid \vec{u}=\vec{w} \quad \text{on} \quad \partial\,\Omega_D\right\}$$

so that it matches the boundary data $\vec{w}$. Now one could easily see that the weak solution need not be twice differentiable, and it does not even need to be continuous.

Any solution of (1.1) and (1.8) is a solution of the weak formulation too. This can be seen from the straightforward construction of (2.7) and (2.4). The question is whether the solution of (2.7) and (2.4) is uniquely defined (up to a constant pressure, if no Neumann boundary conditions are defined).

2.3 Uniqueness of the Weak Solution

In this context it is usual to restate the weak formulation (2.7) and (2.4) using the bilinear forms $a : V_D \times V_D \to \mathbb{R}$ and $b : V_D \times L_2(\Omega) \to \mathbb{R}$:

$$a(\vec{u},\vec{v}) = \int_\Omega \nabla\vec{v} : \tau(\vec{u}), \qquad b(\vec{v},q) = -\int_\Omega p\,\nabla\cdot\vec{v} \tag{2.8}$$

and the continuous functional

$$f(\vec{v}) = \int_\Omega \vec{v}\cdot\rho\vec{g} \;\in\; V' \tag{2.9}$$

where $V = \mathcal{H}^1(\Omega)^d$ and V' is the dual space of V. Here, we deviate from the formulation of Elman et al. (2005) and Dohrmann and Bochev (2004), who use $\nabla\vec{v}$ instead of the full viscosity tensor $\tau(\vec{v})$. Using $\tau(\vec{v})$ is more appropriate in modeling variable viscosity flow. We also deviate from the formulation of Burstedde et al. (2009) and Tabata (2006), who use $\frac{1}{2}(\nabla\vec{u}+\nabla\vec{u}^T)$ instead of $\nabla\vec{u}$ also for the test functions. Olshanskii and Reusken (2006) justify the extension of the results from the analysis of the simpler formulation with $\nabla\vec{u}$ to the more accurate one with $\frac{1}{2}(\nabla\vec{u}+\nabla\vec{u}^T)$. In either case, we can formulate the saddle-point problem:

$$\text{Find } (\vec{u},p)\in(V\times M) \text{ such that:} \qquad \begin{aligned} a(\vec{u},\vec{v}) + b(\vec{v},p) &= f(\vec{v}) \\ b(\vec{u},q) &= 0 \end{aligned} \tag{2.10}$$

Using the kernel space of b,

$$Z = \{\vec{v}\in V \mid b(\vec{v},q) = 0 \quad \forall q\in M\}$$

Bochev and Lehoucq (2006) and Müller (2008) as well as numerous other authors formulate the condition for unique solvability of (2.10), which was developed by Brezzi (1974) in terms of the following theorem:

Theorem 1. *The saddle-point problem* (2.10) *defines an isomorphism* $V_D \times L_2(\Omega) \to V_D' \times L_2'(\Omega)$ *if and only if there exist positive constants* α, β *such that:*

$$a(\vec{v},\vec{v}) \geq \alpha \|\vec{v}\|_{1,\Omega}^2 \quad \forall \vec{v} \in Z \qquad \textit{V-coercivity of } a(\cdot,\cdot) \tag{2.11}$$

$$\inf_{q \in L_2 \neq \text{const}} \sup_{\vec{v} \in V_D \neq 0} \frac{b(\vec{v},q)}{\|\vec{v}\|_{1,\Omega} \|q\|_{0,\Omega}} \geq \beta \quad \textit{inf-sup condition for } b(\cdot,\cdot) \tag{2.12}$$

with the norms

$$\|\vec{v}\|_{1,\Omega}^2 = \int_\Omega \vec{v}\cdot\vec{v} + \nabla\vec{v} : \nabla\vec{v}, \qquad \|q\|_{0,\Omega} = \left\| q - \frac{1}{|\Omega|}\int_\Omega q \right\| \tag{2.13}$$

These conditions hold for the continuous Stokes problem. The pressure norm in (2.13) indicates that the solution of (2.10) is defined up to a constant pressure. One could use the $\mathcal{L}_2$-norm instead and restrict the pressure space to

$$Q = \left\{ p \in \mathcal{L}_2(\Omega) \mid \int_\Omega p = 0 \right\}. \tag{2.14}$$

Tabata and Suzuki (2000) extended the proof of unique solvability to Rayleigh-Bénard convection in a spherical shell with free-slip boundary conditions and later also to variable-viscosity convection (Tabata and Suzuki, 2002; Tabata, 2006). As mentioned before, although they use a slightly different formulation of the bilinear form a, their results should be valid for our formulation as well. The viscosity in their model is supposed to be a continuously differentiable function of position, time and temperature. It is also confined to an interval between extremal values on which the error estimates depend. However, the inf-sup constant β in their model decreases linearly with the overall viscosity contrast.

Olshanskii and Reusken (2006) prove the well-posedness of (2.10) for a discontinuous viscosity with β independent of the mesh size h and the magnitude of the viscosity jump. To get this result, they use the pressure space

$$Q = \left\{ p \in \mathcal{L}_2(\Omega) \mid \int_\Omega \eta^{-1} p = 0 \right\} \tag{2.15}$$

with the scalar product

$$(p,q)_Q = \int_\Omega \eta^{-1} p\, q = (\eta^{-1}p, q) \qquad \forall p,q \in Q \tag{2.16}$$

In (Olshanskii and Reusken, 2004) they also prove an equivalent result for the standard pressure space $\mathcal{L}_2(\Omega)$, but with a norm composed of both, the $\mathcal{L}_2$-norm and (2.16). Without this viscosity dependent pressure norm, no viscosity independent inf-sup constant can be derived.

2.4 Velocity and Pressure Discretization

Eqs. (2.7) and (2.4) are discretized on a uniform square mesh using finite-dimensional subspaces $V_D^h \subset V_D$ and $M^h \subset L_2(\Omega)$. The discrete problem then becomes:

Find $\vec{u}_h \in V_D^h$ and $p_h \in M^h$ such that

$$\int_\Omega \nabla \vec{v}_h : \tau(\vec{u}_h) - \int_\Omega p_h \nabla \cdot \vec{v} = \int_\Omega \vec{v}_h \cdot \rho \vec{g}_h \qquad \forall \vec{v}_h \in V_0^h \tag{2.17}$$

$$\int_\Omega q_h \nabla \cdot \vec{u}_h = 0 \qquad \forall q_h \in M^h \tag{2.18}$$

with V_0^h being the finite-dimensional subspace of V_0, defined in (2.5).

The discrete test and solution functions can be represented as linear combinations of a suitable set of basis functions

$$\vec{u}_h = \sum_{j=1}^{n_i} \mathbf{u}_j \vec{\phi}_j + \sum_{j=n_i+1}^{n_i+n_\partial} \mathbf{u}_j \vec{\phi}_j, \qquad \vec{v}_h = \sum_{j=1}^{n_i} \mathbf{v}_j \vec{\phi}_j \tag{2.19}$$

$$p_h = \sum_{j=1}^{n} \mathbf{p}_j \psi_j, \qquad q_h = \sum_{j=1}^{n} \mathbf{q}_j \psi_j \tag{2.20}$$

where n_i is the number of inner and n_∂ the number of boundary nodes: $n = n_i + n_\partial$. Therefore, (2.17) and (2.18) can be written in block matrix form as a linear equation system

$$\begin{bmatrix} \mathbf{A} & B^T \\ B & 0 \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ g \end{bmatrix} \tag{2.21}$$

with matrix entries

$$\mathbf{a}_{ij} = \int_\Omega \nabla \vec{\phi}_i : \tilde{\nabla} \vec{\phi}_j \cdot \eta_k \psi_k; \qquad b_{kj} = -\int_\Omega \psi_k \nabla \cdot \vec{\phi}_j \tag{2.22}$$

$$\mathbf{f}_i = \sum_{j=1}^{n_i+n_\partial} \rho \mathbf{g}_j \int_\Omega \vec{\phi}_i \cdot \vec{\phi}_j + \sum_{j=n_i+1}^{n_i+n_\partial} \mathbf{u}_j \int_\Omega \nabla \vec{\phi}_i : \nabla \vec{\phi}_j \tag{2.23}$$

$$g_k = \sum_{j=n_i+1}^{n_u+n_\partial} \mathbf{u}_j \int_\Omega \psi_k \nabla \cdot \vec{\phi}_j \tag{2.24}$$

for $i, j = 1 \cdots n_i$ and $k = 1 \cdots n_p$. In the computation of $\mathbf{a}_{ij}$, k is iterated only through the vicinity of i, j, although the formulation is valid for all k. $\tilde{\nabla}$ is the modified gradient to yield the stress tensor which is given in (1.7). We use piecewise bilinear basis functions for both velocity components and pressure, taking the value one at node j and zero at all other nodes on the

mesh. These so-called Q_1–Q_1 elements give the simplest possible globally continuous approximation. The same basis functions are used for the body force $\vec{g}_h$ as well as for the viscosity η in τ:

$$\vec{g}_h = \sum_{j=1}^{n_i+n_\partial} \mathbf{g}_j \vec{\phi}_j, \qquad \eta_h = \sum_{j=1}^{n} \eta_j \psi_j \tag{2.25}$$

The piecewise linear viscosity formulation has already been implemented in the multigrid test framework of Müller (2008).

From now on, in all discrete formulations $\mathbf{u}$ and p denote the finite element coefficient vectors of the discrete velocities $\vec{u}_h$ and pressures p_h.

2.5 Stabilization Using Local Pressure Projections

The discrete inf-sup condition, analogous to (2.12), is:

$$\inf_{q_h \in M^h \neq \text{const}} \sup_{\vec{v}_h \in V_D^h \neq 0} \frac{b(\vec{v}_h, q_h)}{\|\vec{v}_h\|_{1,\Omega} \|q_h\|_{0,\Omega}} \geq \beta > 0 \tag{2.26}$$

with β independent of the grid spacing h. If it is not satisfied by the discrete velocity and pressure functions, additional terms can be added to (2.17) or (2.18) in order to enlarge the numerator or otherwise satisfy (2.26). This is called *stabilization* and it is well known that this is necessary for an equal-order discretization (Elman et al., 2005; Müller, 2008).

Different stabilization techniques for the Q_1–Q_1 element pair have been developed. Dohrmann and Bochev (2004) give an overview and develop a new technique based on local pressure projections. They thereby avoid the use of penalty methods, which relax (disturb) Eq. (2.18). These techniques have been used especially in the variable-viscosity case for computational convenience (e.g. Tabata and Suzuki, 2002). They also avoid residual terms which usually lead to a different Stokes matrix in every iteration step. Bochev et al. (2006), who give error and stability analysis of this technique, introduce a weaker form of the inf-sup condition, which is satisfied also by Q_1–Q_1 and Q1-P0 discretizations:

$$\sup_{\vec{v}_h \in V_D^h \neq 0} \frac{b(\vec{v}_h, q_h)}{\|\vec{v}_h\|_{1,\Omega}} \geq c_1 \|q_h\|_{0,\Omega} - c_2 h \|\nabla q_h\|_{0,\Omega} \qquad \forall q_h \in M^h \neq \text{const} \tag{2.27}$$

Here, c_1 and c_2 are positive constants, independent of h. The last term in (2.27) is called *inf-sup deficiency*. With a suitable projection operator $\Pi : L^2(\Omega) \to R_0$ from the (piecewise linear) pressure space onto the piecewise constant space

$$R_0 = \{p_h \in L^2(\Omega) \quad | \quad p_h|_{\Omega_e} \in \mathcal{P}_0(\Omega_e) \quad \forall \Omega_e \in \mathcal{T}_h\} \tag{2.28}$$

the deficiency can be estimated by

$$h\|\nabla p_h\| \leq c_3\|p_h - \Pi p_h\| \quad \forall p_h \in M^h \tag{2.29}$$

and then the subtraction of

$$c(p,q) = \int_\Omega \frac{1}{\eta}(p - \Pi p)(q - \Pi q) \tag{2.30}$$

from the left side of (2.4) gives a stable formulation with another matrix C entering the lower right block of the Stokes matrix in (2.21). The projection Π itself is implicitly defined by the condition

$$\int_{\Omega_e} (\Pi p - p) = 0 \quad \forall \Omega_e \in \mathcal{T}_h \tag{2.31}$$

and can therefore be calculated locally. ($\mathcal{T}_h$ is the tessellation of the problem domain into polygons, in our case quadrilaterals.) Dohrmann and Bochev (2004) give the following element-wise construction of the stabilization matrix:

$$c_{e\_ij} = \frac{1}{\eta_e}\left[\int_{\Omega_e} \psi_i(x)\psi_j(x) - \frac{1}{|\Omega_e|}\int_{\Omega_e}\psi_i(x)\int_{\Omega_e}\psi_j(x)\right] \quad \forall \Omega_e \in \mathcal{T}_h \tag{2.32}$$

where the indices i and j include all the basis functions which do not vanish on the element Ω_e. The global stabilization matrix C is easily assembled from the element matrices and the Stokes system (2.21) then becomes

$$\begin{bmatrix} \mathbf{A} & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ g \end{bmatrix} \tag{2.33}$$

Note, that the first integral in (2.32) gives the local pressure mass matrix M and that the sparsity of C is comparable to that of $\mathbf{A}$ and B. In (2.32) the viscosity is assumed to be element-wise constant. Using a piecewise linear viscosity would complicate the integration because it would inhibit the use of precomputed integrals on the non-adaptive computational mesh. Moreover, it would lead to a C that heavily penalizes smooth pressure functions when the local viscosity contrast is high.

In this formulation a constant pressure belongs to the null space of the operator C when constant viscosity is used. However, as the pressure solution in our examples as well as in mantle convection simulations is not a constant function, the solution p yields a non-vanishing contribution $-Cp$ to the mass equation and therefore violates the incompressibility constraint. Therefore, the stabilization we use is not a consistent one, i.e. the solution of (2.33) is not a solution of (2.21). However, as seen in Section 2.6, in Example 1 it adds approximately 20% to the discretization error of the mass equation, and when using the η-dependent pressure norm (2.16) the stabilization would be consistent.

2.6 Discretization Errors

Bochev et al. (2006) derive the error bound

$$\|\nabla(\vec{u}-\vec{u}_h)\| + \|p-p_h\|_{0,\Omega} \leq ch\left(\|D^2\vec{u}\| + \|D^1 p\|\right) \tag{2.34}$$

with $(\vec{u}, p)$ being the solution of (2.7) and (2.4) and $(\vec{u}_h, p_h)$ the solution of (2.33) on a rectangular grid $\mathcal{T}_h$ using Q_1–Q_1 elements. c is a constant, h the longest edge length of $\mathcal{T}_h$, and D^1 and D^2 are the sums of the squares of the first and second derivatives, respectively. This bound indicates that the error is proportional to the edge length h of the quadratic elements we use. So the influence of C in (2.33) decreases with finer grids and is proportional to h,

As Elman et al. (2005) compute the errors in (2.34) for our Example 1, I compared the deviation of $(\vec{u}_h, p_h)$ from the discrete approximation of the true solution $(\tilde{\vec{u}}_h, \tilde{p})$. This deviation exists because of the above-mentioned inconsistency of the stabilization but it should at least not be much larger than the deviation from the continuous solution. To reach the discretization error limit, the system (2.33) is solved with a solver from Chapter 3 until the relative residual falls below 10^{-12}, regardless the number of iterations. The errors given by Elman et al. (2005) for $l = 3 \ldots 6$ are extrapolated to $l = 7 \ldots 8$. The errors in Table 2.1 indicate

Table 2.1: Discretization errors for Example 1: $h = 2^{1-l}$

l	$\|\nabla(\vec{u}-\vec{u}_h)\|$	$\|p-p_h\|$	$\|\nabla(\tilde{\vec{u}}_h-\vec{u}_h)\|$	$\|\tilde{p_h}-p_h\|$	n_u
3	8.542×10^{0}	7.940×10^{0}	3.448×10^{0}	8.882×10^{0}	98
4	4.124×10^{0}	2.500×10^{0}	1.202×10^{0}	2.907×10^{0}	450
5	2.021×10^{0}	7.533×10^{-1}	3.960×10^{-1}	9.045×10^{-1}	1922
6	1.001×10^{0}	2.248×10^{-1}	1.294×10^{-1}	2.787×10^{-1}	7938
7	$\approx 5 \times 10^{-1}$	$\approx 7 \times 10^{-2}$	4.286×10^{-2}	8.707×10^{-2}	32258
8	$\approx 2.5 \times 10^{-1}$	$\approx 2 \times 10^{-2}$	1.444×10^{-2}	2.789×10^{-2}	130050

that $\vec{u}_h$ gets much closer to $\tilde{\vec{u}}_h$ as to $\vec{u}$, i.e. the stabilization inconsistency introduces very little deviation compared to the discretization of $\vec{u}$ itself. p_h, however, is a bit further "away" from $\tilde{p}_h$ than from p, suggesting that an extra, yet small, deviation from p is introduced by the stabilization.

2.7 Spectral Properties of the Stokes Matrix

As the behavior of iterative solvers is highly dependent of the spectral properties of the underlying matrices, analytic estimates and, where possible, also computations of their eigenvalues are provided.

For the eigenvalues of $\mathbf{A}$ Elman et al. (2005) give the estimate

$$ch^2 \leq \lambda(\mathbf{A}) \leq C, \tag{2.35}$$

where c and C are independent of h, i.e. the largest eigenvalue remains constant. This is confirmed by computing the eigenvalues in our setting.

To enable an efficient iterative solution of the whole Stokes system, the Schur complement $S = B\mathbf{A}^{-1}B + C$ is of crucial importance. The smallest eigenvalue (except the one zero for constant pressures) of $M^{-1}S$ gives an algebraic equivalent of the inf-sup constant δ. (M is the pressure mass-matrix, see (2.32).) Elman et al. (2005) call δ a generalized inf-sup constant because it characterizes the stabilized discrete Stokes system instead of the original one. They give the estimate

$$\delta^2 \leq \lambda(M^{-1}S) \leq 2, \tag{2.36}$$

and when δ is bounded away from zero independently of h, the formulation is stable. (M and S are called spectrally equivalent if one can give constants on the left and right of (2.36).) The computed eigenvalues for constant viscosity, which are given in Table 2.2, show good stability, which is exactly the predicted behavior. As proposed by Elman et al. (2005, p. 275) the condition number of $M^{-1}(B\mathbf{A}^{-1}B+\alpha C)$ can be slightly improved be choosing $\alpha < 1$, although the general behavior doesn't change. The best choice, considering also the asymptotic behavior of λ_{min}, seems to be $\alpha = 0.5$. However, the computations in Chapter 3 use $\alpha = 1$ and α is excluded from the set of parameters to vary. This is also recommended by Elman et al. (2005) who suggest choosing α slightly larger than the value which minimizes the condition number.

To demonstrate the importance of the stabilization, Table 2.2 also shows the extremal nonzero eigenvalues of the un-stabilized formulation. (In this case $M^{-1}S$ has 8 zero eigenvalues.) As the smallest nonzero eigenvalue is $\sim h^2$, iterative solvers will slow down significantly in this case when the grid is refined.

Table 2.2: Extremal eigenvalues of $M^{-1}(B\mathbf{A}^{-1}B + \alpha C)$ for constant viscosity

	$\alpha = 1$		$\alpha = 0.5$		$\alpha = 0.25$		$\alpha = 0$	
l	λ_{min}	λ_{max}	λ_{min}	λ_{max}	λ_{min}	λ_{max}	λ_{min}	λ_{max}
3	0.2213	1.1219	0.1870	0.6453	0.2474	0.4984	0.0119	0.4760
4	0.2001	1.1290	0.1782	0.6554	0.2334	0.5073	0.0032	0.4943
5	0.1881	1.1327	0.1727	0.6593	0.2178	0.5102	0.0008	0.4986
6	0.1803	1.1335	0.1691	0.6604	0.2045	0.5109	0.0002	0.4997

2.8 Variable Viscosity

To model mantle dynamics in the Earth and other terrestrial planets, different types of viscosity variations must be considered. These include exponential variations arising from temperature variations and viscosity jumps associated with phase boundaries, subducting slabs and rising plumes. Therefore, in SSST we investigate the three structures shown in

Figure 2.3. These are similar to models SOLKY, SOLCX and SINKER of May and Moresi (2008). However, our high viscosity inclusions I04–I12 differ from SINKER by having a circular boundary, making it more challenging to discretize and to solve on a uniform rectangular grid. Because this case cannot be properly represented on the coarser grids, Figure 2.3 uses a higher resolution for I04. This requirement of higher resolution has also to be taken into account when examining iterative solvers.

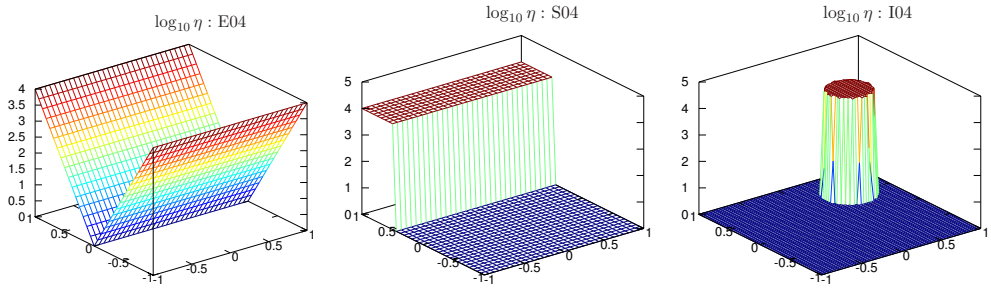

Figure 2.3: Logarithmic plots of viscosity structures E04, S04, I04

Our viscosity models are given entirely as analytic functions, so that iterative solutions of (2.33) can be compared with analytic ones. Viscosity steps are represented using the error function:

$$\text{S04:} \quad \eta = 5000(\text{erf}(1000(y-0.5)+2)+1)+1 \tag{2.37}$$

$$\text{I04:} \quad \eta = 5000(\text{erf}(1000(1/16-(x-0.5)^2-(y-0.5)^2)+2)+1)+1 \tag{2.38}$$

The first factor in (2.37) and (2.38) is one half of the viscosity contrast, which is 10^4 for cases S04 and I04. This convention is extended to 10^8 and 10^{12} for all models and is is indicated by last two digits in the model name. The factor 1000 in (2.37) and (2.38) is chosen as small as possible for the viscosity jump to occur between adjacent grid points also on the finest used grid. There is no loss in generality by setting $\eta_{min} = 1$ as the momentum equation (2.7) can always be divided by η_{min} together with the calculated pressure. Because (2.33) is dominated by the high viscosities in $\mathbf{A}$ which also control the size of the residual, the velocities in Examples 1 and 2 are divided by 3, 6 or 9 orders of magnitude, corresponding to the overall viscosity contrast of 4, 8 or 12 orders of magnitude, respectively.

The viscosity variations affect the spectral properties of $\mathbf{A}$ as expected (see Table 2.3). To be spectrally equivalent to the Schur complement, the pressure mass-matrix now also must account for the viscosity variations. Its local contributions are

$$m_{e\_ij} = \frac{1}{\eta_e}\int_{\Omega_e} \psi_i(x)\psi_j(x) \quad \forall \Omega_e \in \mathcal{T}_h \tag{2.39}$$

with viscosity treated as element-wise constant. Eigenvalues of this pressure mass-matrix scaled Schur complement are given in Table 2.4. They

Table 2.3: Extremal eigenvalues of **A** for constant and variable viscosity

	const		E04		S04		I04	
l	λ_{min}	λ_{max}	λ_{min}	λ_{max}	λ_{min}	λ_{max}	λ_{min}	λ_{max}
4	0.1120	7.8857	2.2561	23011	0.1534	74758	0.1266	62356
5	0.0282	7.9711	0.5334	36452	0.0371	78488	0.0306	74795
6	0.0071	7.9928	0.1306	48680	0.0091	79601	0.0075	78751

Table 2.4: Extremal eigenvalues of $M^{-1}(B\mathbf{A}^{-1}B + 0.5C)$ for variable viscosity

	E04		S04		I04		S12	
l	λ_{min}	λ_{max}	λ_{min}	λ_{max}	λ_{min}	λ_{max}	λ_{min}	λ_{max}
4	0.0527	0.8134	0.1676	0.8966	0.0384	0.9240	0.1676	0.8968
5	0.0485	0.8233	0.1674	0.9402	0.0881	0.9633	0.1674	0.9404
6	0.0475	0.8253	0.1659	0.9658	0.1074	0.9829	0.1659	0.9660

are essentially independent of the magnitude of the viscosity variation because the reciprocal of η is used in (2.32) and (2.39) to build C and M. When solving this scaled Schur complement system, the pressure error is minimized in the M-norm. To reduce the error in this norm, the iteration number of a Krylov method should be independent of the viscosity variation. To have also the L_2-norm of the pressure error reduced below a fixed tolerance, in the worst case the M-norm of the error has to be further reduced by the global magnitude of the viscosity variation. This is explored further in Section 3.2.3.

Chapter 3

Solution of the Stokes Equations in 2D

This chapter describes the iterative solvers which are implemented in SSST for solving the finite-element discretized Stokes equations with strongly varying viscosity. The choices for solvers and preconditioners are based on the spectral properties of the stabilized Q_1–Q_1 discretization that were described in Sections 2.5 and 2.7. With 3-D applications in mind, direct solution of (2.33) is not feasible and is therefore not studied here, although direct methods are indeed used in SSST to compute the eigenvalues on coarse grids (see Sections 2.7 and 3.1). As the Stokes matrix

$$K = \begin{bmatrix} \mathbf{A} & B^T \\ B & -C \end{bmatrix} \tag{3.1}$$

is symmetric, but indefinite, a CG method cannot be used without modification to solve (2.33). One possibility is to segregate the system (2.33) into smaller subsystems for $\mathbf{u}$ and p which are solved separately (Verfürth, 1984). Another possibility is to use a method to solve (2.33) that does not require positive definiteness. One such method is the Krylov subspace method MINRES (Elman et al., 2005) with, for example , a block diagonal preconditioner containing a multigrid algorithm. Applying multigrid as a solver for the whole Stokes system (Tackley, 2008) is advantageous if the spectrum of the Schur complement S grows rapidly on finer grids, thus requiring less iterations on coarser grids. However, this is not the case for the stabilized $Q_1 - Q_1$ discretization, since the spectrum of S does not depend on the number of grid points when preconditioned with the viscosity-scaled pressure mass matrix M_η (see Table 2.4). Another solver, circumventing the indefiniteness of K has been proposed by Bramble and Pasciak (1988). It uses a block-triangular preconditioner to (2.33) leading to a system with a matrix that is positive definite in a nonstandard inner product. This can be solved with a standard CG algorithm. Therefore three solvers for the variable viscosity Stokes system are examined:

- *Pressure Correction (PC):* This is the "classical" segregated algorithm

that is used in TERRA and other mantle convection codes (e.g. CitComS (Zhong et al., 2007)). It solves the Schur complement equation with CG and updates velocity simultaneously.

• *MINRES:* This is a Krylov method for symmetric but indefinite systems. It is parameter free and, mostly with a multigrid preconditioner for $\mathbf{A}$, widely used in recent implementations of Stokes solvers (Burstedde et al., 2009).

• *Bramble-Pasciak CG (BPCG):* It applies CG to a positive-definite reformulation of the Stokes system. It is used also in elasticity models (Meyer and Steidten, 2001; Meyer and Steinhorst, 2005).

The solution for $\mathbf{u}$, i.e. the application of A^{-1}, is done for all solvers in the same way to get a valid comparison between them. As the block matrices $\mathbf{A}, B, C$ are assembled in sparse format, storing three vectors for rows, columns and values, the solvers can exploit the sparsity of the resulting Stokes system.

3.1 Scaling

Because the matrices $\mathbf{A}$ and S have very high condition numbers which depend both on the mesh parameter and on the viscosity contrast, one should scale the system (2.33) prior to solving it as

$$\begin{bmatrix} \mathbf{X} & 0 \\ 0 & Y \end{bmatrix} \begin{bmatrix} \mathbf{A} & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} \mathbf{X} & 0 \\ 0 & Y \end{bmatrix} \begin{bmatrix} \mathbf{X}^{-1} & 0 \\ 0 & Y^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{X} & 0 \\ 0 & Y \end{bmatrix} \begin{bmatrix} \mathbf{f} \\ g \end{bmatrix} \tag{3.2}$$

and then solve it for

$$\begin{bmatrix} \mathbf{u}_s \\ p_s \end{bmatrix} = \begin{bmatrix} \mathbf{X}^{-1} & 0 \\ 0 & Y^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} \tag{3.3}$$

The scaled matrices are denoted

$$\begin{bmatrix} \mathbf{A}_s & B_s^T \\ B_s & C_s \end{bmatrix} = \begin{bmatrix} \mathbf{X} & 0 \\ 0 & Y \end{bmatrix} \begin{bmatrix} \mathbf{A} & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} \mathbf{X} & 0 \\ 0 & Y \end{bmatrix} \tag{3.4}$$

$\mathbf{X}$ and Y should be easy to invert matrices and their squares should be spectrally equivalent to $\mathbf{A}$ and B. The easiest and in most cases close to best choice are the diagonal matrices with entries

$$\mathbf{x}_{ii} = 1/\sqrt{\mathbf{a}_{ii}}, \qquad y_{jj} = 1/\sqrt{m_{jj}}, \tag{3.5}$$

where m_{jj} refer to diagonal elements of the viscosity-scaled pressure mass matrix. As these scaling matrices are symmetric positive definite, the scaled Stokes matrix retains its symmetry. Positive definiteness of the scaling matrices is indeed necessary here. Hence, one can consider this scaling as preconditioning, and the resulting system can be solved with the same methods as the original one. The extremal nonzero eigenvalues of

Table 3.1: Extremal eigenvalues of **XAX** for constant and variable viscosity

	const		E04		S04		I04	
l	λ_{min}	λ_{max}	λ_{min}	λ_{max}	λ_{min}	λ_{max}	λ_{min}	λ_{max}
3	0.1089	1.8895	0.3764	1.6373	0.1531	1.8391	0.0326	1.9600
4	0.0280	1.9714	0.1165	1.8916	0.0383	1.9611	$3.2e-5$	2.0230
5	0.0070	1.9927	0.0305	1.9721	0.0093	1.9916	$6.1e-6$	2.0119
6	0.0018	1.9982	0.0077	1.9930	0.0023	2.0012	$1.2e-6$	2.0058

Table 3.2: Extremal eigenvalues of $Y(B\mathbf{A}^{-1}B+C)Y$ for variable viscosity

	E04		S04		I04		S12	
l	λ_{min}	λ_{max}	λ_{min}	λ_{max}	λ_{min}	λ_{max}	λ_{min}	λ_{max}
3	0.1717	1.2821	0.2500	1.4157	$5.7e-4$	1.3037	0.2500	1.4159
4	0.1220	1.6440	0.2500	1.5747	0.1200	1.6935	0.2500	1.6992
5	0.1101	1.7988	0.2500	1.7765	0.2058	1.9883	0.2500	1.9158
6	0.1072	1.8430	*sing*	*sing*	*sing*	*sing*	*sing*	*sing*

the scaled matrices **XAX** and YSY are given in Tables 3.1 and 3.2. We do not consider the zero eigenvalue of YSY because Elman (1996) and others have pointed out that it does not influence the convergence properties of an iterative solver, i.e., to get closer to the solution, the solver will never add a constant pressure only. Within the solver, yet another preconditioner for **A** is required to yield h-independent iteration numbers, whereas for S one can do without further preconditioning. Here, diag $(M_\eta)^{-1}$, being much cheaper to apply than M_η^{-1}, also reduces the condition number significantly. This is in agreement with the recommendation to use Y^2 from (3.5) as preconditioner for the Schur complement (Elman et al., 2005; May, 2009) of the Stokes equations. A comparison between scaling and mass-matrix preconditioning for the Schur complement will be given in Chapter 5 for the 3D-spherical discretization.

In the following, solution algorithms are always applied to the scaled system (3.2), although the subscripts are omitted for convenience of notation.

3.2 Pressure Correction Algorithm

A very popular algorithm for solving the discretized Stokes equations is the pressure correction PC algorithm, which was first described and analyzed by Verfürth (1984). It decouples the variables by transforming the system (2.33) to

$$\begin{bmatrix} \mathbf{A} & B^T \\ 0 & B\mathbf{A}^{-1}B^T + C \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ B\mathbf{A}^{-1}\mathbf{f} - g \end{bmatrix} \tag{3.6}$$

For an initial pressure a velocity is calculated to fulfill the momentum equation. Then the positive definite Schur complement equation

$$(B\mathbf{A}^{-1}B^T + C)\, p = B\mathbf{A}^{-1}\mathbf{f} - \mathbf{g} \tag{3.7}$$

is solved with a conjugate gradient method. The velocity is updated simultaneously because $\mathbf{A}^{-1}B^T \triangle p$ is available when the residual of (3.7) is updated as shown in Algorithm 1. Sometimes this method is called an Uzawa algorithm, especially in the mantle convection community (Zhong et al., 2007). However, as Uzawa (1958) used stationary iterations as well as a gradient method to solve (3.7), the term "Uzawa method" often denotes a stationary iteration on the Schur complement equation (3.7). Therefore Algorithm 1 is named PC here.

Algorithm 1: Pressure correction: Standard mode

Choose p_0
Solve $\mathbf{A}\mathbf{u}_0 = \mathbf{f} - B^T p_0$ for $\mathbf{u}_0$ until $\frac{||\mathbf{f}-\mathbf{A}\mathbf{u}_0 - B^T p_0||}{||\mathbf{f}-B^T p_0||} < utol$
Compute residual $r_0 = B\mathbf{u}_0 - Cp_0 - g$
for *i=1* **to** N **do**
  if *i=1* **then**
    $s_1 = r_0$
  else
    $\delta = \frac{\langle r_{i-1}, r_{i-1}\rangle}{\langle r_{i-2}, r_{i-2}\rangle}$
    $s_i = r_{i-1} + \delta s_{i-1}$
  end
  Solve $\mathbf{A}\mathbf{v}_i = B^T s_i$ for $\mathbf{v}_i$ until $\frac{||\mathbf{A}\mathbf{v}_i - B^T s_i||}{||B^T s_i||} < vtol$
  $\alpha = \frac{\langle r_{i-1}, r_{i-1}\rangle}{\langle s_i, B\mathbf{v}_i + Cs_i\rangle}$
  $p_i = p_{i-1} + \alpha s_i$
  $\mathbf{u}_i = \mathbf{u}_{i-1} - \alpha \mathbf{v}_i$
  $r_i = r_{i-1} - \alpha(B\mathbf{v}_i + Cs_i)$
  if $\frac{||r_i||}{||r_0||} < ptol$ **then** Exit loop
end

3.2.1 Error Propagation and Solver Restart

Having the velocity corrections as a by-product of the pressure iterations is an advantage of this algorithm, but the drawback is that the application of $\mathbf{A}^{-1}$ has to be done very accurately to retain precision during the iterations, see also Elman (1996). Therefore May and Moresi (2008) impose the requirement on the inner tolerance

$$vtol \leq ptol \tag{3.8}$$

Because in practice this requirement is often difficult to fulfill, Verfürth (1984) proposes to restart this algorithm every 10-20 iterations when velocity solutions are not accurate enough. This idea is the key to run PC

also with inner accuracy as low as the accuracy of the **A**-preconditioner in a coupled solver, which makes a comparison between them feasible. Therefore, in SSST, beside the standard PC algorithm with the pressure correction loop executed only once, also a restarted version, PC.R, is implemented and shown as Algorithm 2.

Algorithm 2: Pressure correction: Restarted mode

Choose p_0
for *j=1* **to** N_o **do**
  $u_0acc = \frac{||\mathbf{f}-\mathbf{A}\mathbf{u}_0-B^Tp_0||}{||\mathbf{f}||}$; $u_0tol = \max(vtol, utol/u_0acc)$
  Solve $\mathbf{A}\mathbf{u}_0 = \mathbf{f} - B^Tp_0$ for $\mathbf{u}_0$ until $\frac{||\mathbf{f}-\mathbf{A}\mathbf{u}_0-B^Tp_0||}{||\mathbf{f}-B^Tp_0||} < u_0tol$
  Compute residual $r_0 = B\mathbf{u}_0 - Cp_0 - g$
  $p_0acc = \frac{||r_0||}{||g||}$;
  $stol = \max(\min(vtol * ptol/utol, 0.5), \min(ptol/p_0acc, 0.8))$
  for *i=1* **to** N_i **do**
    if *i=1* **then**
      $s_1 = r_0$
    else
      $\delta = \frac{\langle r_{i-1}, r_{i-1}\rangle}{\langle r_{i-2}, r_{i-2}\rangle}$
      $s_i = r_{i-1} + \delta s_{i-1}$
    end
    Solve $\mathbf{A}\mathbf{v}_i = B^Ts_i$ for $\mathbf{v}_i$ until $\frac{||\mathbf{A}\mathbf{v}_i-B^Ts_i||}{||B^Ts_i||} < vtol$
    $\alpha = \frac{\langle r_{i-1}, r_{i-1}\rangle}{\langle s_i, B\mathbf{v}_i+Cs_i\rangle}$
    $p_i = p_{i-1} + \alpha s_i$
    $\mathbf{u}_i = \mathbf{u}_{i-1} - \alpha\mathbf{v}_i$
    $r_i = r_{i-1} - \alpha(B\mathbf{v}_i + Cs_i)$
    if $\frac{||r_i||}{||r_0||} < stol$ **then** Exit loop
  end
  $\mathbf{u}_0 = \mathbf{u}_i$; $p_0 = p_i$
  if $\frac{||\mathbf{f}-\mathbf{A}\mathbf{u}_i-B^Tp_i||}{||\mathbf{f}||} < utol \wedge \frac{||g-B\mathbf{u}_i+Cp_i||}{||g||} < ptol$ **then** Exit loop
end

The outer loop, however, is not restarted after a fixed number of iterations because this would lead to another parameter to choose, namely, this fixed iteration number. Instead, in each outer loop *stol* is set to the maximum of *vtol* and the remaining residual reduction needed to get the relative residual of the mass equation below *ptol*. Hence, the condition (3.8) is met in every loop in PC.R, the residuals of momentum and mass equation decrease equally, and PC.R gives very accurate solutions even with low accuracy of the inner solution. Therefore, the standard PC algorithm will not considered further.

Another optimization of PC.R is the reduced accuracy *utol* in the $\mathbf{u}_0$-computation. As it makes no sense to have a very accurate velocity solution being polluted by yet inexact corrections in early PC calls, velocity

accuracy should increase simultaneously with pressure accuracy. In later PC cycles, $\mathbf{u}_0$ may even happen to be already within desired accuracy, in this case its recomputation is skipped. The PC loop, however, is always executed as the Schur complement residual can already fulfill its stopping criterion while the the mass equation residual does not. So, an upper limit for the residual reduction in every PC call is set, dependent on whether *vtol* is so high that the choice of *stol*, based on worst case error estimates, would lead to an unacceptable high value (0.5) or *ptol* has already reached (0.8). Further considerations on the stopping tolerances are given in Section 3.2.3.

It should be mentioned that, although developed differently, PC.R is essentially the same method as MGUZAWA of Peters et al. (2005), except that they use multigrid as inner solver. They name the stationary iteration, the outer loop, after Uzawa and call the inner loop a preconditioner, for which they choose a CG algorithm on the Schur complement, which is the PC algorithm when velocity is updated simultaneously.

3.2.2 Inner Solver

As we now have three solver levels, the term "inner solver" still refers to the evaluations of $\mathbf{A}^{-1}$ which is indeed the innermost level and computationally the most intensive. When, as a measure of solver performance, inner iterations are counted (see Section 3.5), these are also the iterations of the $\mathbf{A}$-solver. The performance and robustness of a well-configured PC.R still depends on the inner solver. For convenience, the first choice for the inner solver is a CG algorithm. This is possible because $\mathbf{A}$ is symmetric and positive definite. The comparison of the Stokes solvers in this study is done with the same CG used as inner solver, which is also called a preconditioner for $\mathbf{A}$ when a coupled solver is used. As the extremal eigenvalues in Tables 2.3 and 3.1 show, the condition number $\kappa(\mathbf{A})$ is roughly proportional to the number of grid points. This suggests the use of multigrid as inner solver which utilizes the much smaller condition number on coarser grids. This is also consensus of many researchers (Verfürth, 1984; Elman, 1996; Elman et al., 2005; Zhong et al., 2007; Burstedde et al., 2009) and would have been our choice as well. However, multigrid is not easy to implement in the variable viscosity case. Because the focus here is a comparison between different outer solvers, and because of its simpler error estimates, we therefore chose to use a CG here as the inner solver.

3.2.3 Stopping Criteria and Error Norms

While the conjugate gradient method on $A\mathbf{x} = \mathbf{b}$ minimizes the A-norm of the error $||\mathbf{e}||_A$, unfortunately this quantity is not accessible without knowing the solution in advance. What is known is the relationship between

error and residual after an arbitrary number k of CG-iterations:

$$\frac{||\mathbf{e}_k||_A}{||\mathbf{e}_0||_A} \leq \sqrt{\kappa(A)}\frac{||\mathbf{r}_k||}{||\mathbf{r}_0||} \tag{3.9}$$

Note that this is not a sharp estimate as it holds also in the worst case. Mostly error reduction goes faster than this. Tables 3.1 and 3.2 show $\sqrt{\kappa(\mathbf{A})} \sim 1/h$. Therefore, we need $||\mathbf{r}_k||/||\mathbf{r}_0|| \sim h$ for the velocity error staying constant as grid resolution increases. In contrast, for the scaled Schur complement (3.7) $\sqrt{\kappa(S)} \sim 1$, independently of h. Considering that the discretization error is proportional to h (see Table 2.1), so should be the iteration error as well. This introduces yet another $\sim h$ factor to the stopping tolerances for the relative residual norms. To optimize the algorithm, we now need 3 stopping tolerances:

$ptol \sim h$	Schur complement equation (outer iteration: pressure)
$utol \sim h^2$	Momentum equation (initial velocity computation)
$vtol \sim h$	Momentum equation (inner iteration: vel. search direction)

Although the velocity search directions are computed using the same $\mathbf{A}$, *vtol* does not have the same h-proportionality as *utol*, because the inner iterations need not be performed until the discretization-error level is reached. It is sufficient to have the inner iteration error remain constant with finer grid resolution. The three tolerances are derived in the following way:

$$ptol = 16h * tol/\sqrt{\kappa(S_5)} \tag{3.10}$$

$$utol = 256h^2 * tol/\sqrt{\kappa(\mathbf{A}_5)} \tag{3.11}$$

$$vtol = \min\left(16h * itol * \sqrt{\frac{\kappa(S_5)}{\kappa(\mathbf{A}_5)}}, 0.1\right) \tag{3.12}$$

A_5 and S_5 are the scaled matrices for $l = 5$, where $h = 1/16$ in Example 1 implies the pre-factor 16. *tol* is the desired error reduction for $l = 5$. Both, *tol* and *itol* are varied by powers of 10 to get the iteration error norms within 1% deviation from the lowest possible value which is determined by the discretization error. Table 3.3 gives the condition numbers of $\mathbf{A}$ and S for all viscosity profiles when $l = 5$ and the largest possible choices for the input values *tol* and *itol* to get the iteration error norms within 1% deviation from the lowest possible value. It will be seen in Section 3.5 that often a very low inner accuracy suffices. Indeed, in many cases, the calculated stopping tolerance *vtol* remains at its upper limit of 0.1 on all but the finest grids. This limit, which was introduced to prevent *vtol* from getting too large on coarse grids, could also be set as high as 0.5, but 0.1 gives slightly better results and works for also for the coupled solvers.

Especially when applying high magnitudes of absolute viscosity variations, it is of crucial importance which norm is used to measure the

Table 3.3: Condition numbers and stopping tolerances for PC.R

Visc.	$\kappa(\mathbf{A}_5)$	$\kappa(S_5)$	tol_M	tol_{L_2}	$itol$
E04	65	16	10^{-5}	10^{-5}	1
E08	21	55	10^{-3}	10^{-5}	1
E12	10	99	10^{-3}	10^{-6}	1
S04	217	7	10^{-3}	10^{-5}	1
S08	217	7	10^{-3}	10^{-5}	1
S12	217	7	10^{-3}	10^{-6}	1
I04	$3.3 \cdot 10^5$	9.5	10^{-2}	10^{-2}	1
I08	$3 \cdot 10^9$	10.5	10^{-2}	10^{-2}	1
I12	$3 \cdot 10^{13}$	10.5	1	1	10^{-2}

pressure error. In most such cases one usually one wants not only the M-norm, but also the L_2-norm to be as low as possible. Then stopping criteria cannot be derived by (3.9) alone. As is seen in Table 3.3, with the profiles E12 and S12, the mass equation residual must be reduced by another three orders of magnitude to get not only the M-norm but also the L_2-norm of the error as close as possible to the discretization error. Such a need for a further error reduction can be confirmed also theoretically. In the worst case, residual thresholds derived from (3.9) must be divided by the square root of the global viscosity contrast, since this is scaled out by M, which always depends on η here. However, the examples we considered indicate that in most cases this condition can be substantially weakened (see also Table 3.3). Note that all these considerations for $ptol, utol, vtol$ and their derivations are valid also for the coupled solvers. Only the values of tol and $itol$ are solver-specific (see Section 3.5).

3.2.4 Convergence Properties

The convergence number ρ of an iterative method is given theoretically as

$$\rho = \lim_{k \to \infty} \frac{||\mathbf{e}_k||_A}{||\mathbf{e}_0||_A}. \tag{3.13}$$

For a CG method it is well known (Golub and Van Loan, 1996) that

$$\rho \sim \frac{\sqrt{\kappa(A)} - 1}{\sqrt{\kappa(A)} + 1}. \tag{3.14}$$

Therefore, we have in the PC algorithm

$$\rho \sim \frac{\sqrt{\kappa(S)} - 1}{\sqrt{\kappa(S)} + 1}. \tag{3.15}$$

if the inner system is solved either exactly or in a way that the matrix $\tilde{\mathbf{A}}$, which describes the inverse of the inexact application of $\mathbf{A}^{-1}$, is spectrally

equivalent to $\mathbf{A}$. This is the case when the above-mentioned stopping criteria with $vtol \sim h$ are applied . The constant $itol$ in (3.12) determines the proportionality factor in (3.15).

3.3 Preconditioned MINRES Algorithm

In order to avoid the need to obtain a very accurate inner solution for the pressure correction algorithm and because a restarted algorithm is often considered an adequate poor man's approach, recent implementations of convection models often favor the preconditioned MINRES algorithm to solve the discrete Stokes equations. Moreover, MINRES has the appeal of not requiring any parameters for making the algorithm efficient. Larin and Reusken (2008) found MINRES to be the most robust algorithm with respect to changing values in the (constant) viscosity parameter compared to inexact Uzawa and coupled multigrid methods. Also Elman et al. (2005) recommend preconditioned MINRES to solve the discretized Stokes equations. Therefore, it is also included in SSST as a candidate for the most suitable solver. The MINRES algorithm was developed by Paige and Saunders (1975) and is applied directly to the symmetric but indefinite system (2.33) which is reformulated to $K\mathbf{x} = \mathbf{b}$ following (3.1). Let Q be a preconditioner for K. The algorithm which is used here and outlined in Algorithm 3 is taken from Elman et al. (2005).

Algorithm 3: Preconditioned MINRES.

$\mathbf{v}_0 = \mathbf{w}_0 = \mathbf{w}_1 = 0$
Choose $\mathbf{x}_0$; compute $\mathbf{r}_1 = \mathbf{b} - K\mathbf{x}_0$
Solve $Q\mathbf{z}_1 = \mathbf{r}_1$; set $\gamma_1 = \sqrt{\langle \mathbf{z}_1, \mathbf{r}_1 \rangle}$
Set $\eta = \gamma_1,\ s_0 = s_1 = 0,\ c_0 = c_1 = 1$
for *i=1* **to** N **do**
  $\mathbf{z}_i = \mathbf{z}_i / \gamma_i$
  $\delta_i = \langle K\mathbf{z}_i, \mathbf{z}_i \rangle$
  $\mathbf{r}_{i+1} = K\mathbf{z}_i - (\delta_i/\gamma_i)\mathbf{r}_i - (\gamma_i/\gamma_{i-1})\mathbf{r}_{i-1}$
  Solve $Q\mathbf{z}_{i+1} = \mathbf{r}_{i+1};\ \gamma_{i+1} = \sqrt{\langle \mathbf{z}_{i+1}, \mathbf{r}_{i+1} \rangle}$
  $\alpha_0 = c_i\delta_i - c_{i-1}s_i\gamma_i$
  $\alpha_1 = \sqrt{\alpha_0^2 + \gamma_{i+1}^2}$
  $\alpha_2 = s_i\delta_i + c_{i-1}c_i\gamma_i$
  $\alpha_3 = s_{i-1}\gamma_i$
  $c_{i+1} = \alpha_0/\alpha_1;\ s_{i+1} = \gamma_{i+1}/\alpha_1$
  $\mathbf{w}_{i+1} = (\mathbf{z}_i - \alpha_3\mathbf{w}_{i-1} - \alpha_2\mathbf{w}_i)/\alpha_1$
  $\mathbf{x}_i = \mathbf{x}_{i-1} + c_{i+1}\eta\mathbf{w}_{i+1}$
  $\eta = -s_{i-1}\eta$
  if $|\eta| < ptol$ **then** Exit loop
end

3.3.1 Preconditioning and Convergence Properties

Q should be an easy to invert close approximation to K. The block diagonal preconditioner, proposed by Elman et al. (2005), with blocks spectrally equivalent to the momentum operator $\mathbf{A}$ and to the Schur complement S, respectively, is adapted to our problem as follows:

$$Q = \begin{bmatrix} \mathbf{A}_0 & 0 \\ 0 & T \end{bmatrix} \approx \begin{bmatrix} \mathbf{A} & 0 \\ 0 & B\mathbf{A}^{-1}B^T + C \end{bmatrix} \tag{3.16}$$

As we have the scaled matrices here, see (3.4), we can estimate eigenvalues from Tables 3.1 and 3.2, and choosing T to be the identity matrix I is sufficient. The preconditioned MINRES algorithm minimizes the residual in the norm

$$||\mathbf{r}||_{Q^{-1}} = \sqrt{\langle Q^{-1}\mathbf{r}, \mathbf{r}\rangle} \tag{3.17}$$

To relate this to the error $\mathbf{e} = \mathbf{x}_k - \mathbf{x}$, we need the following estimates which must hold for all vectors of the respective spaces:

$$\underline{\alpha} \leq \frac{\langle \mathbf{A}\mathbf{u}, \mathbf{u}\rangle}{\langle \mathbf{A_0}\mathbf{u}, \mathbf{u}\rangle} \leq \overline{\alpha} \tag{3.18}$$

$$\underline{\gamma}^2 \leq \frac{\langle (B\mathbf{A}^{-1}B^T + C)p, p\rangle}{\langle Mp, p\rangle} \leq \overline{\gamma}^2 \tag{3.19}$$

$$\underline{\delta} \leq \frac{\langle Mp, p\rangle}{\langle p, p\rangle} \leq \overline{\delta} \tag{3.20}$$

$$\Upsilon \leq \frac{\langle Mp, p\rangle}{\langle Cp, p\rangle} \tag{3.21}$$

All these spectral equivalences with their estimated bounds are used to derive the inclusion intervals for the eigenvalues of $Q^{-1}K$ (see Elman et al., 2005):

$$[-a, -b] \cup [c, d] = \left[-\overline{\delta}^2(1+\Upsilon), \frac{1}{2}\left(\underline{\alpha} - \sqrt{\underline{\alpha}^2 + 4\underline{\alpha}\underline{\gamma}^2\underline{\delta}^2}\right)\right] \cup \left[\underline{\alpha}, \overline{\alpha} + \overline{\gamma}^2\overline{\delta}^2\right] \tag{3.22}$$

It is desirable to have all these quantities independent of the mesh-parameter h. Note that M is already scaled, so $\underline{\delta}\underline{\gamma}^2$ and $\overline{\delta}\overline{\gamma}^2$ are independent of h and can be estimated from Table 3.2. Moreover, from Wathen (1987) we know that in the worst case we have $\underline{\delta} = 1/4$ and $\overline{\delta} = 9/4$. For the stabilization used here, $\Upsilon = 1$. Now the only requirement for deriving mesh-independent convergence bounds is that also $\underline{\alpha}$ and $\overline{\alpha}$ are independent of h. With $\mathbf{P} = \mathbf{I}$ this would not be the case as the lowest eigenvalue of $\mathbf{A}$ is $\sim h^2$. When the action of $\mathbf{P}^{-1}$ is approximated by an iterative solver, then the quantities $\underline{\alpha}$ and $\overline{\alpha}$ are determined by this solution process and can be estimated by its residual reduction. If, see Section 3.2.2, a CG method is used here, the stopping tolerance *vtol* is also taken from (3.12).

The value of *itol* can be varied, but (3.12) suggests $itol = 1$ as a good choice to start with. So we have

$$\frac{\overline{\alpha}}{\underline{\alpha}} \approx \frac{\overline{\delta}\overline{\gamma}^2}{\underline{\delta}\underline{\gamma}^2}, \tag{3.23}$$

and we also have a wide overlap of the intervals $[\underline{\alpha}, \overline{\alpha}]$ and $\left[\underline{\delta}\underline{\gamma}^2, \overline{\delta}\overline{\gamma}^2\right]$ independent of h. This, in turn, suggests that also $[a, b]$ and $[c, d]$ overlap, because the matrices $\mathbf{A}$ and S are scaled, with the median of their eigenvalues close to 1. However, this overlap might be improved by varying *itol*, leading to an optimal residual reduction of the MINRES algorithm, which is:

$$||\mathbf{r}_{2k}||_{Q^{-1}} \leq 2\left(\frac{\sqrt{ad}-\sqrt{bc}}{\sqrt{ad}+\sqrt{bc}}\right)^k ||\mathbf{r}_0||_{Q^{-1}} \tag{3.24}$$

The error $\mathbf{e}$ is measured in the norm induced by the matrix

$$E = \begin{bmatrix} \mathbf{A} & 0 \\ 0 & M \end{bmatrix}, \tag{3.25}$$

i.e., it "adds" the $\mathbf{A}$-norm for velocity and the M-norm for pressure. These are exactly the norms in which the errors are reduced in the pressure correction algorithm. Now the whole error reduction can be estimated to

$$c_1||\mathbf{e}||_E \leq ||\mathbf{r}||_{Q^{-1}} \leq c_2||\mathbf{e}||_E \tag{3.26}$$

if $\underline{\alpha}$ and $\overline{\alpha}$ are independent of h. Elman et al. (2005) derive the bounds

$$c_1 = \underline{\alpha}\underline{\gamma}^2\left(1 + \frac{1}{2}\underline{\delta}^2\underline{\gamma}^2/\underline{\alpha} - \sqrt{1 + \frac{1}{4}\underline{\delta}^4\underline{\gamma}^4/\underline{\alpha}^2}\right), \tag{3.27}$$

$$c_2 = \max\left\{2\overline{\alpha} + \overline{\delta}^2\overline{\gamma}^2, 2\overline{\alpha\gamma}^2\right\}. \tag{3.28}$$

Therefore, the residual $||\mathbf{r}||_{Q^{-1}}$ can be reduced using the same stopping tolerance *ptol* as in the pressure correction algorithm up to a constant, and we have the convergence number

$$\rho \sim \frac{\sqrt{ad}-\sqrt{bc}}{\sqrt{ad}+\sqrt{bc}}, \tag{3.29}$$

which is minimized by applying the considerations on *vtol* and *itol* given above. As the error bound (3.26) is not as tight as (3.9), the constant *tol* should be chosen smaller than in PC.

3.4 Bramble-Pasciak-CG

Another possibility for solving (2.33), given by Bramble and Pasciak (1988), is to use a block triangular preconditioner and define an inner product in

which the resulting system is symmetric positive definite. This can then be solved with a CG algorithm. The use of a block triangular instead of a block diagonal preconditioner is also proposed by Geenen et al. (2009), who found that solving such a preconditioned system with GMRES needs half the iterations compared to solving the block diagonal preconditioned system with MINRES. The use of a CG method, however, is attractive from the computationally point of view, because implementation and error estimates are simple and well understood.

The block triangular preconditioner, which Bramble and Pasciak (1988) propose, comprises a preconditioner $\mathbf{A}_0$ for $\mathbf{A}$, for which

$$\alpha_0\langle\mathbf{Au},\mathbf{u}\rangle \leq \langle\mathbf{A}_0\mathbf{u},\mathbf{u}\rangle \leq \alpha_1\langle\mathbf{Au},\mathbf{u}\rangle \qquad \forall\mathbf{u}\in\mathcal{H}^1(\Omega)^d \tag{3.30}$$

holds with $\alpha_1 < 1$ and α_0 as large as possible, preferably independent of h. Then the whole preconditioner

$$K_0 = \begin{bmatrix} \mathbf{A}_0^{-1} & 0 \\ B^T\mathbf{A}_0^{-1} & -I \end{bmatrix} \tag{3.31}$$

is multiplied from the left to (2.33), leading to the system

$$\mathcal{M}\begin{bmatrix}\mathbf{u}\\ p\end{bmatrix} = \begin{bmatrix} \mathbf{A}_0^{-1}\mathbf{A} & \mathbf{A}_0^{-1}B^T \\ B\mathbf{A}_0^{-1}(\mathbf{A}-\mathbf{A}_0) & B\mathbf{A}_0^{-1}B^T + C \end{bmatrix}\begin{bmatrix}\mathbf{u}\\ p\end{bmatrix} = \begin{bmatrix} \mathbf{A}_0^{-1}\mathbf{f} \\ B\mathbf{A}_0^{-1}\mathbf{f} - g \end{bmatrix} \tag{3.32}$$

which is positive definite when the following inner product on the space $V_D \times L_2(\Omega)$ is used:

$$\left\langle \begin{bmatrix}\mathbf{u}\\ p\end{bmatrix}, \begin{bmatrix}\mathbf{v}\\ s\end{bmatrix} \right\rangle = \langle\mathbf{Au},\mathbf{v}\rangle - \langle\mathbf{A_0u},\mathbf{v}\rangle + \langle p, s\rangle \tag{3.33}$$

Therefore, (3.32) is solved with a standard CG method, given in Algorithm 4. The original method of Bramble and Pasciak (1988) differs from standard CG in the computation of the weighting factor β for updating the search direction and is less efficient by introducing a third $\mathbf{A}$-inversion per iteration. The lower efficiency could also be reproduced in our Example 1, and for that reason this method is not examined further. But also in Algorithm 4, quantities are always recomputed when possible to reduce accumulation of round-off errors and of errors arising from inexact $\mathbf{A}_0^{-1}$-evaluations. In particular, in every iteration $\mathbf{A}_0^{-1}$ is applied twice, even though $\mathbf{x}$ can be reused to update $\mathbf{v}$. Therefore, a fast version of BPCG, introduced by Peters et al. (2005), where the cost per iteration is reduced as much as possible, has been implemented. It is given as Algorithm 5. In this fast version, however, the error of $\mathbf{x}$ also enters t and via $\mathbf{x}$ and t into α as well. Therefore, the update term $\alpha\mathbf{x}$ of $\mathbf{v}$ in Algorithm 5 has a larger error than the recomputation in Algorithm 4 gives. The relative error increase in Example 1 is about 3-30, diminishing as the true solution is approached. For that reason, Algorithm 5 is always applied in

Algorithm 4: Bramble-Pasciak CG.

Choose $\mathbf{u}_0, p_0$
Compute residual $\bar{\mathbf{v}}_0 = \mathbf{f} - \mathbf{A}\mathbf{u}_0 - B^T p_0$
Solve $\mathbf{A}_0\mathbf{v}_0 = \bar{\mathbf{v}}_0$ for $\mathbf{v}_0$ until $\frac{||\mathbf{A}_0\mathbf{v}_0 - \bar{\mathbf{v}}_0||}{||\bar{\mathbf{v}}_0||} < vtol$
Compute residual $r_0 = \delta(B(\mathbf{u}_0 + \mathbf{v}_0) - Cp_0 - g)$
$\mathbf{w}_0 = \mathbf{v}_0, \quad s_0 = r_0, \quad \mathbf{z}_0 = \binom{\mathbf{v}_0}{r_0}$
$zz_0 = \mathbf{v}_0^T\mathbf{A}\mathbf{v}_0 - \bar{\mathbf{v}}_0^T\mathbf{v}_0 + s_0^T s_0/\delta$
for *i=1* **to** N **do**
  $\bar{\mathbf{x}} = \mathbf{A}\mathbf{w}_{i-1} + B^T s_{i-1}$
  Solve $\mathbf{A}_0\mathbf{x} = \bar{\mathbf{x}}$ for $\mathbf{x}$ until $\frac{||\mathbf{A}_0\mathbf{x} - \bar{\mathbf{x}}||}{||\bar{\mathbf{x}}||} < vtol$
  $t = \delta(B(\mathbf{x} - \mathbf{w}_{i-1}) + Cs_{i-1})$
  $\alpha = zz_{i-1}/(\mathbf{x}^T\mathbf{A}\mathbf{w}_{i-1} - \bar{\mathbf{x}}\mathbf{w}_{i-i} + t^T s_{i-i})$
  $\mathbf{u}_i = \mathbf{u}_{i-1} + \alpha\mathbf{w}_{i-1}$
  $p_i = p_{i-1} + \alpha s_{i-1}$
  Compute residual $\bar{\mathbf{v}}_i = \mathbf{f} - \mathbf{A}\mathbf{u}_i - B^T p_i$
  Solve $\mathbf{A}_0\mathbf{v}_i = \bar{\mathbf{v}}_i$ for $\mathbf{v}_i$ until $\frac{||\mathbf{A}_0\mathbf{v}_i - \bar{\mathbf{v}}_i||}{||\bar{\mathbf{v}}_i||} < vtol$
  Compute residual $r_i = \delta(B(\mathbf{u}_i + \mathbf{v}_i) - Cp_i - g)$
  $\mathbf{z}_i = \binom{\mathbf{v}_i}{r_i}, \quad$ **if** $\frac{||\mathbf{z}_i||}{||\mathbf{z}_0||} < ptol$ **then** Exit loop
  $zz_i = \mathbf{v}_i^T\mathbf{A}\mathbf{v}_i - \bar{\mathbf{v}}_i^T\mathbf{v}_i + s_i^T s_i/\delta, \quad \beta = zz_i/zz_{i-1}$
  $\mathbf{w}_i = \mathbf{v}_i + \beta\mathbf{w}_{i-1}$
  $s_i = r_i + \beta s_{i-1}$
end

the restarted mode similar to what is done with the PC algorithm (see Section 3.2.1), and it is named BPCG.R. The criteria for restarting are slightly simpler than for PC.R. The threshold for the relative residual, *bptol*, is set to a fixed value between 0.03 and 0.2, and $N_i = N_o = 30$.

3.4.1 Preconditioner for A and Convergence Properties

As in PC and MINRES, the preconditioner $\mathbf{A}_0$ is not explicitly given but defined by an iterative method which applies $\mathbf{A}^{-1}$ approximately. However, to satisfy $\alpha_1 < 1$ in (3.30), it is $k_{bp}\mathbf{A}$ which is inverted, where $k_{bp} < 1$ is as large as possible, dependent on *vtol*. Roughly speaking, the more exact we apply $\mathbf{A}^{-1}$, the larger k_{bp} can be chosen to be. The value of k_{bp} can be estimated from the condition number of $\mathbf{A}$, given in Table 3.3 for the various viscosity profiles. As *vtol* gives the desired residual reduction, (3.9) describes the (worst case) error reduction. This gives a rough estimate of the largest local error, which defines the upper limit of k_{bp} needed to satisfy (3.30). In case of a highly localized error this upper limit is

$$k_{bp} = 1 - \frac{||\mathbf{e}_k||_A}{||\mathbf{e}_0||_A} \geq 1 - \sqrt{\kappa(A)}\frac{||\mathbf{r}_k||}{||\mathbf{r}_0||} = 1 - \sqrt{\kappa(A)}vtol \approx 1 - 16h*itol\sqrt{\kappa(S_5)}. \tag{3.34}$$

For explanation of the symbols, see the text following (3.12).

Algorithm 5: BPCG.R (avoids recomputations)

Choose $\mathbf{u}_0, p_0$
for *j=1* **to** N_o **do**
  Compute residual $\bar{\mathbf{v}}_0 = \mathbf{f} - \mathbf{A}\mathbf{u}_0 - B^T p_0$
  Solve $\mathbf{A}_0\mathbf{v}_0 = \bar{\mathbf{v}}_0$ for $\mathbf{v}_0$ until $\frac{||\mathbf{A}_0\mathbf{v}_0 - \bar{\mathbf{v}}_0||}{||\bar{\mathbf{v}}_0||} < vtol$
  Compute residual $r_0 = \delta(B(\mathbf{u}_0 + \mathbf{v}_0) - Cp_0 - g)$
  $\mathbf{w}_0 = \mathbf{v}_0, \quad s_0 = r_0, \quad \mathbf{z}_0 = \binom{\mathbf{v}_0}{r_0}$
  if *j==1* **then** Store $\mathbf{z}_{00} = \mathbf{z}_0$
  $bptol = \min(ptol * \mathbf{z}_{00}/\mathbf{z}_0, 0.2)$
  $\hat{\mathbf{v}}_0 = \mathbf{A}\mathbf{v}_0, \quad \hat{\mathbf{w}}_0 = \hat{\mathbf{v}}_0$
  $zz_0 = \hat{\mathbf{v}}_0^T\mathbf{v}_0 - \bar{\mathbf{v}}_0^T\mathbf{v}_0 + s_0^T s_0/\delta$
  for *i=1* **to** N_i **do**
    $\bar{\mathbf{x}} = \hat{\mathbf{w}}_{i-1} + B^T s_{i-1}$
    Solve $\mathbf{A}_0\mathbf{x} = \bar{\mathbf{x}}$ for $\mathbf{x}$ until $\frac{||\mathbf{A}_0\mathbf{x} - \bar{\mathbf{x}}||}{||\bar{\mathbf{x}}||} < vtol$
    $t = \delta(B(\mathbf{x} - \mathbf{w}_{i-1}) + Cs_{i-1})$
    $\alpha = zz_{i-1}/(\mathbf{x}^T\hat{\mathbf{w}}_{i-1} - \bar{\mathbf{x}}\mathbf{w}_{i-i} + t^T s_{i-i})$
    $\mathbf{u}_i = \mathbf{u}_{i-1} + \alpha\mathbf{w}_{i-1}$
    $p_i = p_{i-1} + \alpha s_{i-1}$
    $\bar{\mathbf{v}}_i = \bar{\mathbf{v}}_{i-1} - \alpha\bar{\mathbf{x}}$
    $\mathbf{v}_i = \mathbf{v}_{i-1} - \alpha\mathbf{x}$
    $\hat{\mathbf{v}}_\mathbf{i} = \mathbf{A}\mathbf{v}_i$
    $\mathbf{r}_i = \mathbf{r}_{i-1} - \alpha t$
    $\mathbf{z}_i = \binom{\mathbf{v}_i}{r_i}, \quad$ **if** $\frac{||\mathbf{z}_i||}{||\mathbf{z}_0||} < bptol$ **then** Exit loop
    if $(\hat{\mathbf{v}}_i^T\mathbf{v}_i - \bar{\mathbf{v}}_i^T\mathbf{v}_i < 0)$ **then** Exit loop
    $zz_i = \hat{\mathbf{v}}_i^T\mathbf{v}_i - \bar{\mathbf{v}}_i^T\mathbf{v}_i + s_i^T s_i/\delta, \quad \beta = zz_i/zz_{i-1}$
    $\mathbf{w}_i = \mathbf{v}_i + \beta\mathbf{w}_{i-1}$
    $s_i = r_i + \beta s_{i-1}$
    $\hat{\mathbf{w}}_i = \hat{\mathbf{v}}_i + \hat{\mathbf{w}}_{i-1}$
  end
  if $\frac{||\mathbf{f} - \mathbf{A}\mathbf{u}_i - B^T p_i||}{||\mathbf{f}||} < utol \wedge \frac{||g - B\mathbf{u}_i + Cp_i||}{||\mathbf{g}||} < ptol$ **then** Exit loop
  $\mathbf{u}_0 = \mathbf{u}_i; \quad p_0 = p_i$
end

To derive a reasonable choice of *itol*, let us consider the estimate of the condition number of $\mathcal{M}$ by Meyer and Steidten (2001). It assumes (3.18), which is similar to (3.30), and an equivalence estimate for the Schur complement:

$$\underline{\beta} \leq \frac{\langle (B\mathbf{A}^{-1}B^T + C)p, p\rangle}{\langle p, p\rangle} \leq \overline{\beta}. \tag{3.35}$$

The resulting upper bound of the condition number is then

$$\kappa(\mathcal{M}) \leq 4\frac{\overline{\alpha}}{\underline{\alpha}}\frac{\max(\overline{\alpha}, \overline{\beta})}{\min(\underline{\alpha}, \underline{\beta})}. \tag{3.36}$$

This upper limit of $\kappa(\mathcal{M})$ can clearly be minimized by reducing $\overline{\alpha}/\underline{\alpha}$ only, but it is also desirable that $[\underline{\alpha}, \overline{\alpha}] \subset [\underline{\beta}, \overline{\beta}]$. Therefore, Meyer and Steidten (2001) multiply the second row of K_0 in (3.31) with a pre-factor δ. This shifts the second interval to $[\delta\underline{\beta}, \delta\overline{\beta}]$ to yield an optimal overlap with $\overline{\alpha}/\underline{\alpha}$. In SSST δ has been varied around 1 (between 0.6 and 10), but $\delta = 1$ gives the best results. This is also in accordance with theory. As $\mathbf{A}$ and S are already scaled and have nearly the same median of their eigenvalues, $\delta = 1$ suffices, especially if $itol \leq 0.1$, i.e.$vtol$ is small. (See also the discussion of (3.23).)

In our numerical examples, $itol$ is varied by powers of 10, decreasing from 1. Because k_{bp} would be too low to yield a proper preconditioner when $itol = 1$, a lower threshold, similar to what is done in Bramble and Pasciak (1988, Ex.2), is applied, namely

$$k_{bp} = \max(1 - 16h * itol\sqrt{\kappa(S_5)}, 0.5) \tag{3.37}$$

Although the choice (3.37) yields a positive definite $\mathbf{A} - \mathbf{A}_0$, this positive definiteness can be violated because of round-off errors. This has been observed in BPCG.R as well as in BPCG in some cases, especially when viscosity variations are large. Therefore, the condition $\hat{\mathbf{v}}_i^T\mathbf{v}_i - \bar{\mathbf{v}}_i^T\mathbf{v}_i > 0$ is checked in every iteration step. If it is violated, BPCG is restarted, but without changing k_{bp} or $vtol$. Because then again one has $\hat{\mathbf{v}}_i^T\mathbf{v}_i - \bar{\mathbf{v}}_i^T\mathbf{v}_i > 0$, it should really be a matter of round-off errors. When this necessary condition for positive definiteness is checked, it is possible to increase k_{bp} to get a better convergence. This has been done, and in Section 3.5 results are shown for

$$k_{bp} = \min(\max(1 - 2h * itol\sqrt{\kappa(S_5)}, 0.8), 0.99) \tag{3.38}$$

The actual values of this k_{bp} for different grid refinements and viscosities can be seen in Table 3.4. These are the values for the optimal $itol$ regarding

Table 3.4: Specific values of $k_{bp} = \min(\max(1 - 2h * itol\sqrt{\kappa(S_5)}, 0.8), 0.99)$

Visc.	E04		E12		S04		S12		I04		I12	
l	6	8	6	8	6	8	6	8	6	8	6	8
$Ex.1$	0.975	0.99	0.99	0.99	0.83	0.96	0.83	0.96	0.81	0.95	0.98	0.99
$Ex.2$	0.80	0.94	0.80	0.85	0.83	0.96	0.83	0.96	0.81	0.95	0.81	0.95

performance. Note that if using (3.37), k_{bp} would also be between 0.8 and 0.99 for BPCG.R performing optimal, but one would have to use a smaller $itol$ to get these values, making the overall computational effort larger.

As with PC.R, the condition

$$\frac{||\mathbf{f} - \mathbf{A}\mathbf{u} - B^T p||}{||\mathbf{f}||} < utol \quad \text{and} \quad \frac{||g - B\mathbf{u} + Cp||}{||g||} < ptol. \tag{3.39}$$

is also checked after every BPCG loop and if restart is necessary, $bptol$ is again set to a fixed value between 0.03 and 0.2 for every subsequent loop.

As this moderate residual reduction can be obtained in a few (usually 3-7) iterations, the overall effort to fulfill (3.39) is close to optimal with these settings.

3.5 Results

Table 3.5: Iteration numbers of PC for Example 1

l	4		5		6		7		8	
Visc.	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i
E04	24	149	25	320	30	836	24	1434	25	2907
E08	36	124	46	304	54	711	57	1535	52	3168
E12	41	99	58	258	84	734	75	1344	91	3286
S04	16	200	17	447	19	881	20	2021	22	4367
S08	18	210	20	538	22	1225	25	2524	25	5316
S12	26	296	32	780	36	1774	35	3617	37	7107
I04	18	1269	17	2713	15	4970	16	11299	16	23611
I08	32	1911	27	4453	29	11196	25	20363	21	42827
I12	47	4035	41	7848	37	16407	41	55592	38	84734

Table 3.6: Iteration numbers of MINRES for Example 1

l	4		5		6		7		8	
Visc.	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i
E04	60	366	44	502	59	1491	52	2812	44	5035
E08	73	309	109	728	126	1561	154	4048	98	5769
E12	43	245	81	864	101	2131	97	4207	94	8470
S04	28	240	31	469	37	948	39	2173	40	4730
S08	38	328	54	767	48	1368	48	3117	48	6658
S12	64	577	59	937	63	1795	63	4225	58	8188
I04	54	2463	48	4874	34	7495	32	13929	32	29178
I08	148	9724	76	12514	53	17528	49	33399	49	73200
I12	140	19048	53	16357	53	35938	53	70872	54	150650

All solvers in this study are able to solve the test problems. The difference in computational time, based on the number of inner iterations, varies in most cases by a factor of less than two between the fastest and slowest solver. In this section, tables and plots are shown for Example 1. To give a comparison, Figure 3.3, containing the most significant plots, is also included for Example 2. The other tables and plots for Example 2 are given in Section A.1. All algorithms are run as described in the previous sections.

Tables 3.5, 3.6 and 3.7 show iteration numbers for the algorithms examined. The quantity of interest is the total number of inner iterations, it_i, to solve (3.4). For every solver and every viscosity structure the accuracies of the outer and the inner solver have been optimized to yield a

Table 3.7: BPCG.R iterations: $k_{bp} = \min(\max(1 - 2kh\sqrt{\kappa(S_5)}, 0.8), 0.99)$

l	4		5		6		7		8	
Visc.	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i
E04	27	166	22	295	26	853	23	1483	22	3260
E08	40	153	49	306	61	785	58	1732	42	3038
E12	40	133	50	334	65	961	60	1862	60	4030
S04	24	240	25	501	42	1533	30	2728	25	5257
S08	15	216	16	563	17	1331	18	2939	20	7589
S12	37	416	47	945	42	1897	41	3929	37	8981
I04	20	1263	16	2079	17	4663	19	12561	24	31658
I08	51	5396	39	8232	32	14971	29	28353	24	50053
I12	75	14118	32	12510	32	26881	27	46403	27	94809

Table 3.8: Stopping criteria for all solvers in Example 1

Solver	PC.R		MINRES		BPCG.R	
Visc.	*tol*	*itol*	*tol*	*itol*	*tol*	*itol*
E04	10^{-5}	10	10^{-5}	0.1	10^{-4}	0.1
E08	10^{-6}	1	10^{-5}	0.1	10^{-4}	0.1
E12	10^{-6}	1	10^{-6}	0.001	10^{-5}	0.01
S04	10^{-5}	10	10^{-4}	1	10^{-4}	1
S08	10^{-5}	10	10^{-6}	1	10^{-3}	0.1
S12	10^{-7}	10	10^{-8}	1	10^{-6}	1
I04	0.01	0.1	10^{-4}	1	1	1
I08	0.1	1	10^{-6}	1	10	1
I12	0.1	1	10^{-8}	0.1	10	0.1

low it_i (see also Section 3.2.3). Therefore, the number of outer iterations, it_o, varies with each solver but stays roughly independent of grid size in all cases. To easily compare it_o and it_i between the solvers for all grid levels and viscosities, line and bar graphs are provided in Figures 3.1, 3.2 and 3.3.

3.5.1 Inner Accuracy

Table 3.8 shows the stopping criteria of each solver and viscosity structure. The inner accuracy *vtol* is defined by *itol* in (3.12). If $itol = 10$, the upper threshold $vtol = 0.1$ is applied on all grid levels. It is also applied for E08 and E12 if $itol = 1$. Note that Table 3.8 does not show the highest possible *itol* but the optimal one. With only little loss in performance, BPCG.R could be run with $itol = 1$ for all viscosity structures, except I12, the most challenging one. For MINRES this would also be true if it is run in restarted mode (MINRES.R). Then, as for PC.R, for E- and S- structures the upper limit of $vtol = 0.1$ would already be sufficient, whereas MINRES and BPCG.R need $vtol = 0.03$ to get a stable solution for S-structures. Because $\mathbf{A}$ is almost singular to machine precision for

the high viscosity inclusion, the calculated and estimated eigenvalues in Table 3.1 are much higher. Then *vtol* ranges from 10^{-4} to 10^{-8} with the same high *itol*, leading to a significant increase in it_i for all solvers.

It is also of interest to consider how the solvers' performance depend on the inner accuracy, especially if a very efficient **A**-solver, such as multigrid, is available. To investigate this, I compared the increase of it_i when *itol* is reduced by 10^2 compared to the optimal value. For PC.R the relative increase is about 1.5 for E- and S-structures and about 1.2 for I-structures. For MINRES, where the optimal choice of *itol* often coincides with the highest possible choice, it is ≈ 1.5 for E-, ≈ 2.2 for S- and ≈ 1.4 for I-structures. For BPCG.R it is ≈ 1.8 for E- and S- and ≈ 1.4 for I-structures. However, as *itol* could also be increased for BPCG.R, it has a range of 2-3 orders of magnitude without changing it_i by more than 1.4 for all viscosity structures. To summarize, the performance of PC.R and BPCG.R depends weakly on *itol*, and that of MINRES a little more. This gives an advantage to the restarted algorithms for practical computations, where it is often difficult to define an optimal inner accuracy, and one instead relies upon a built-in safety factor.

3.5.2 Viscosity Variations

Another important question is: How do the iteration numbers depend on the viscosity contrast $\Delta\eta$? The three types of viscosity variations each yield a somewhat different behavior.

For exponential variations, the inner iteration count it_i is almost independent of $\Delta\eta$ for PC.R and BPCG.R and depends only mildly on $\Delta\eta$ for MINRES. However, the outer iteration count it_o depends strongly on $\Delta\eta$ for every solver, although PC.R and BPCG.R use a smaller *itol* for E12 than for E04. At first this seems surprising, because it implies that fewer inner iterations per outer iteration are needed when the viscosity contrast is increased. But Figures 3.5 and 3.6 show that the residual for E12 does not decrease monotonically. The reason may be that in this case the error-residual relation is closer to the worst-case estimate (3.9). The increased iteration numbers of MINRES may be caused by the single call of the algorithm. A restarted version, MINRES.R, needs $\{it_i, it_o\} = \{65, 4020\}$ for E04 and $\{it_i, it_o\} = \{229, 6712\}$ for E12 on the finest grid level with $itol = 1$. It is still worse than the other solvers, both, in terms of iteration number it_i and of dependence on $\Delta\eta$, but more efficient than MINRES.

For the single viscosity step, both, it_o and it_i, increase mildly with viscosity contrast for all solvers. The S-structures show the smallest differences between the three solvers with PC.R having slightly smaller it_i and $\Delta\eta$-dependence than the others.

For the high viscosity inclusion, BPCG.R and especially MINRES have increased iteration numbers on the coarsest mesh. This may be due to the inaccurate representation of the high viscosity "disc" on the rectangular elements. But on finer grids, numbers grow almost asymptotically, as

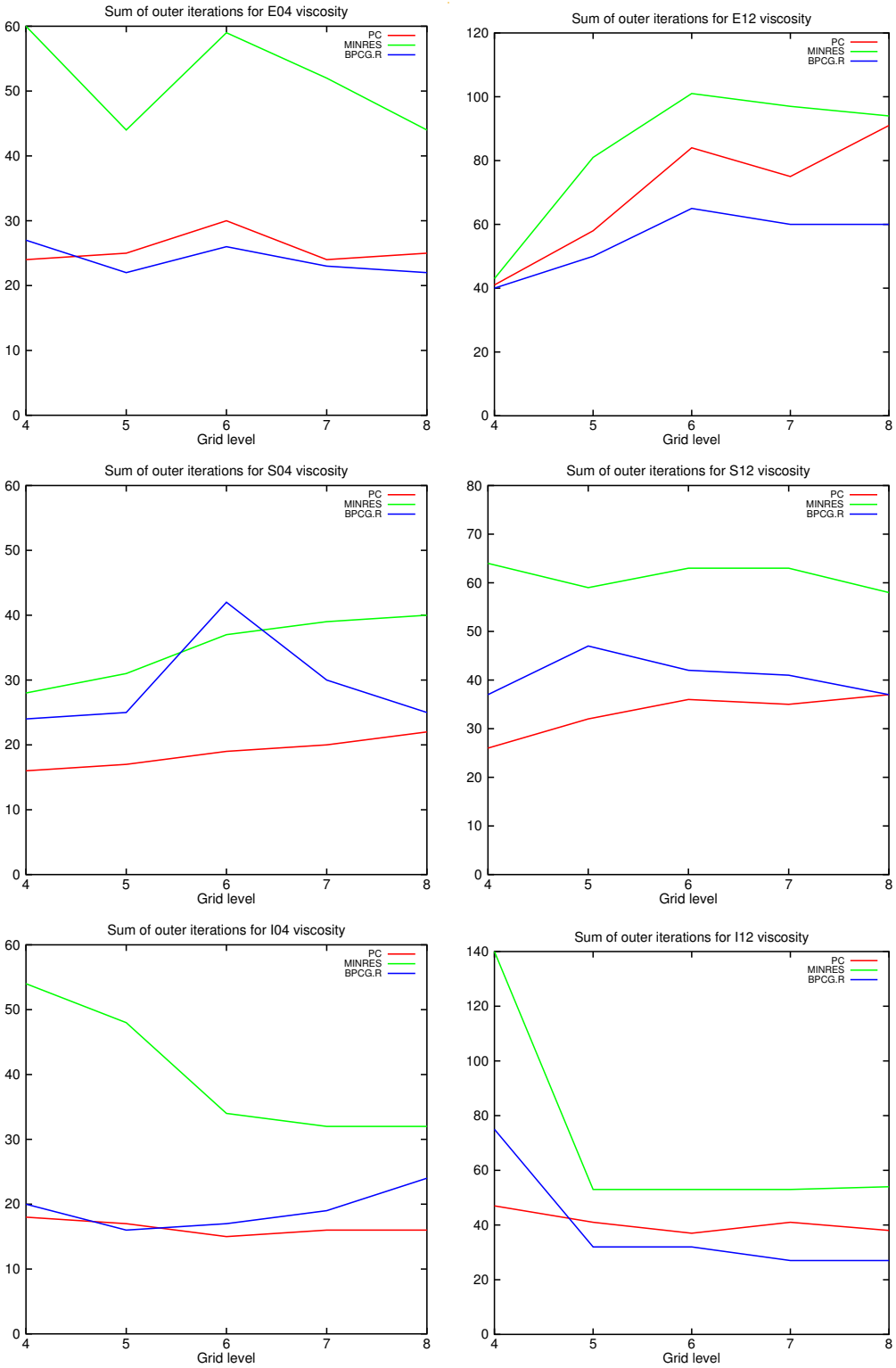

Figure 3.1: Sum of outer iterations for all solvers in Example 1

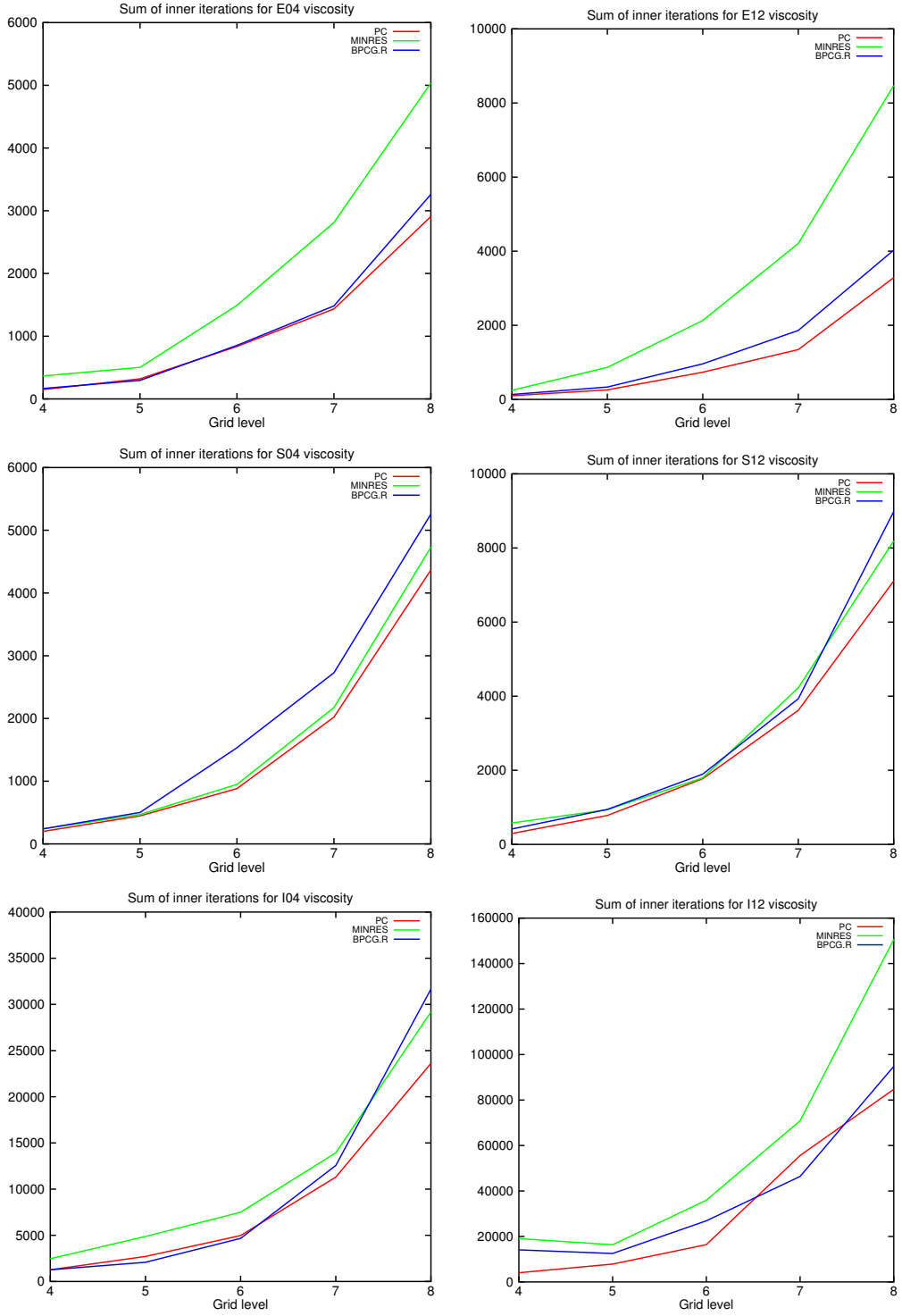

Figure 3.2: Sum of inner iterations for all solvers in Example 1

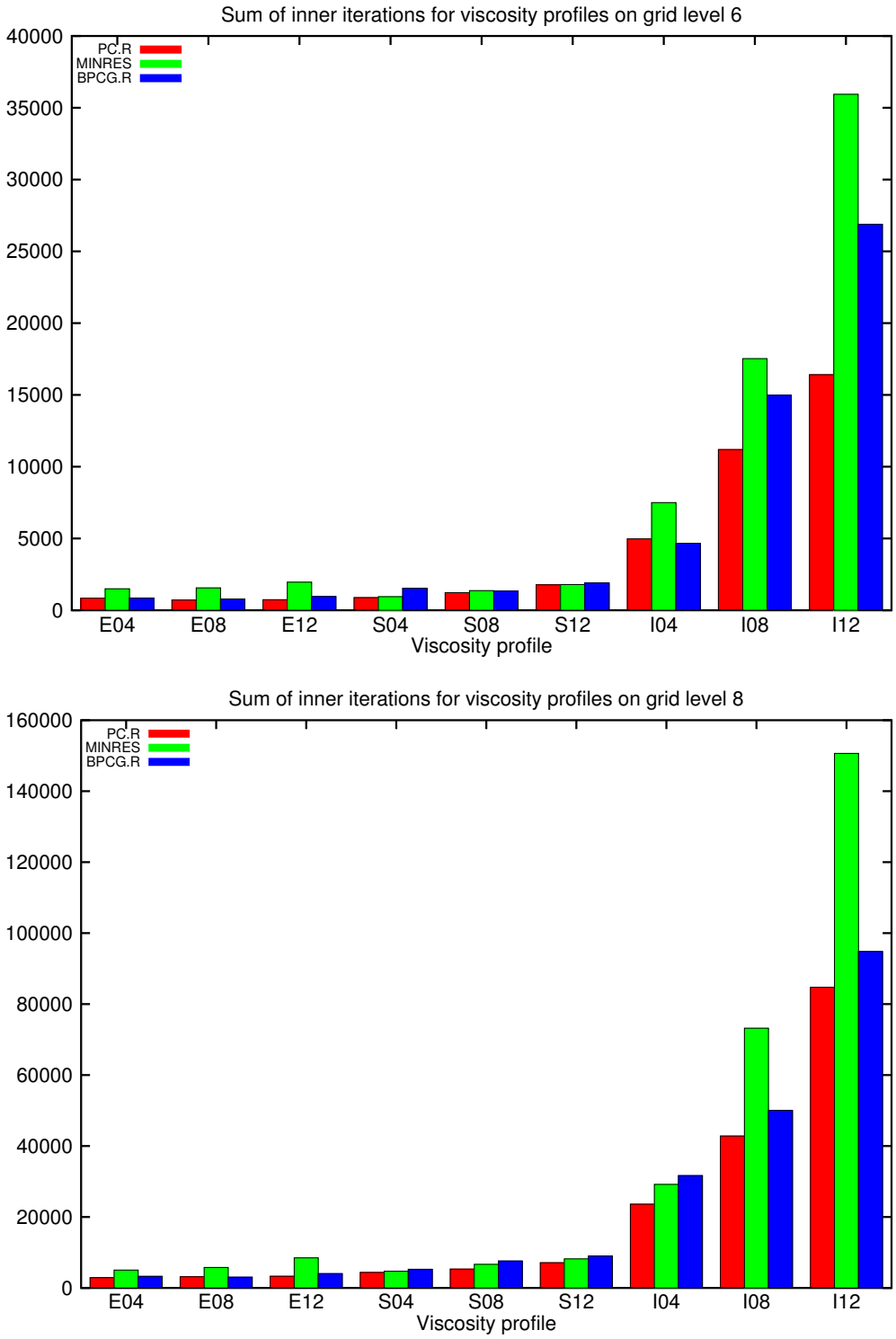

Figure 3.3: Sum of inner iterations for Example 1

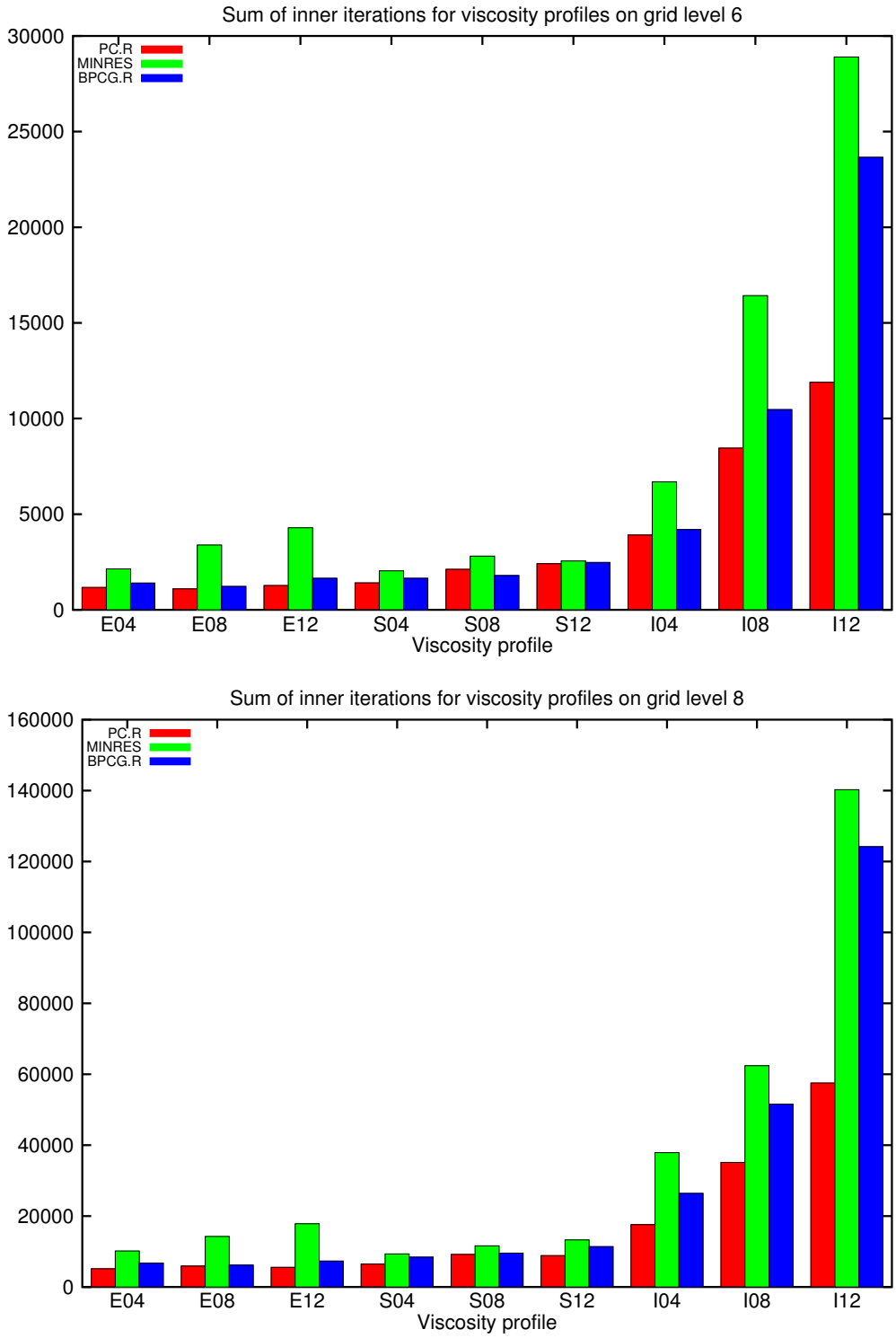

Figure 3.4: Sum of inner iterations for Example 2

expected. With constant $itol$, it_o stays roughly constant, while it_i doubles with every increase of $\Delta\eta$. Again, as with exponential profiles, MINRES is worse than the restarted solvers by about 1.5. Especially with high $\Delta\eta$ on fine grids, PC.R is the fastest solver. The advantage over the other solvers is more prominent in Example 2, see Figure 3.4. As mentioned before, I12 is very challenging as it is difficult to get the inner solution accurate enough, especially on the finest grid. Therefore, iteration numbers can increase a bit more than expected.

3.5.3 Convergence Properties and Residual Reduction

As an example, the residual reduction of all solvers is shown in Figures 3.5 and 3.6 for Example 1 with the exponential viscosity structures E04 and E12. The curves are not drawn into one diagram, because every solver minimizes another quantity: PC the S-norm of the p-error, MINRES the Q^{-1}-norm of the (u,p)-residual, and BPCG the $\mathcal{M}$-norm of the (u,p)-error in the inner product (3.33), see Sections 3.2, 3.3 and 3.4. MINRES is therefore the only method expected to have a monotonically decreasing residual. PC.R and BPCG.R should yield a low-frequency monotonically residual reduction with high-frequency oscillations. When they hit an eigenvector of a large eigenvalue, the residual may also drop significantly.

The observed residual reduction for the three solvers is as expected. In PC.R the restarts prevent the curve from flattening out which is often observed when doing many PC-iterations at once. MINRES shows an almost linear residual reduction over many iterations, but sometimes no residual reduction occurs at odd iteration steps. This is due to a symmetry of eigenvalues of $Q^{-1}K$ around their midpoint and can also be predicted analytically (Elman et al., 2005, p. 307). BPCG.R shows a residual reduction quite similar to that of PC.R, but with distinct high-frequency oscillations, because matrix and inner product describing the error-residual relation are more complicated than in PC.R. Again, the "long-term" reduction is almost linear.

3.6 Discussion and Bibliographical Notes

The performance and robustness of PC.R may be surprising with the low inner accuracy in these computations. However, compared to the residual reduction in every PC loop, the inner accuracy is high enough to fulfill (3.8). Moreover, with PC it is possible to apply well suited and highly optimized solvers to the pressure and velocity subsystems, making PC.R the most efficient method for variable-viscosity Stokes systems. MINRES, in this study, is competitive regarding efficiency for handling a single viscosity step. This could be enhanced a little by restarting MINRES. The results of MINRES.R are not shown here, because they differ only slightly from those of MINRES: MINRES.R is faster for E- and S-structures and slower

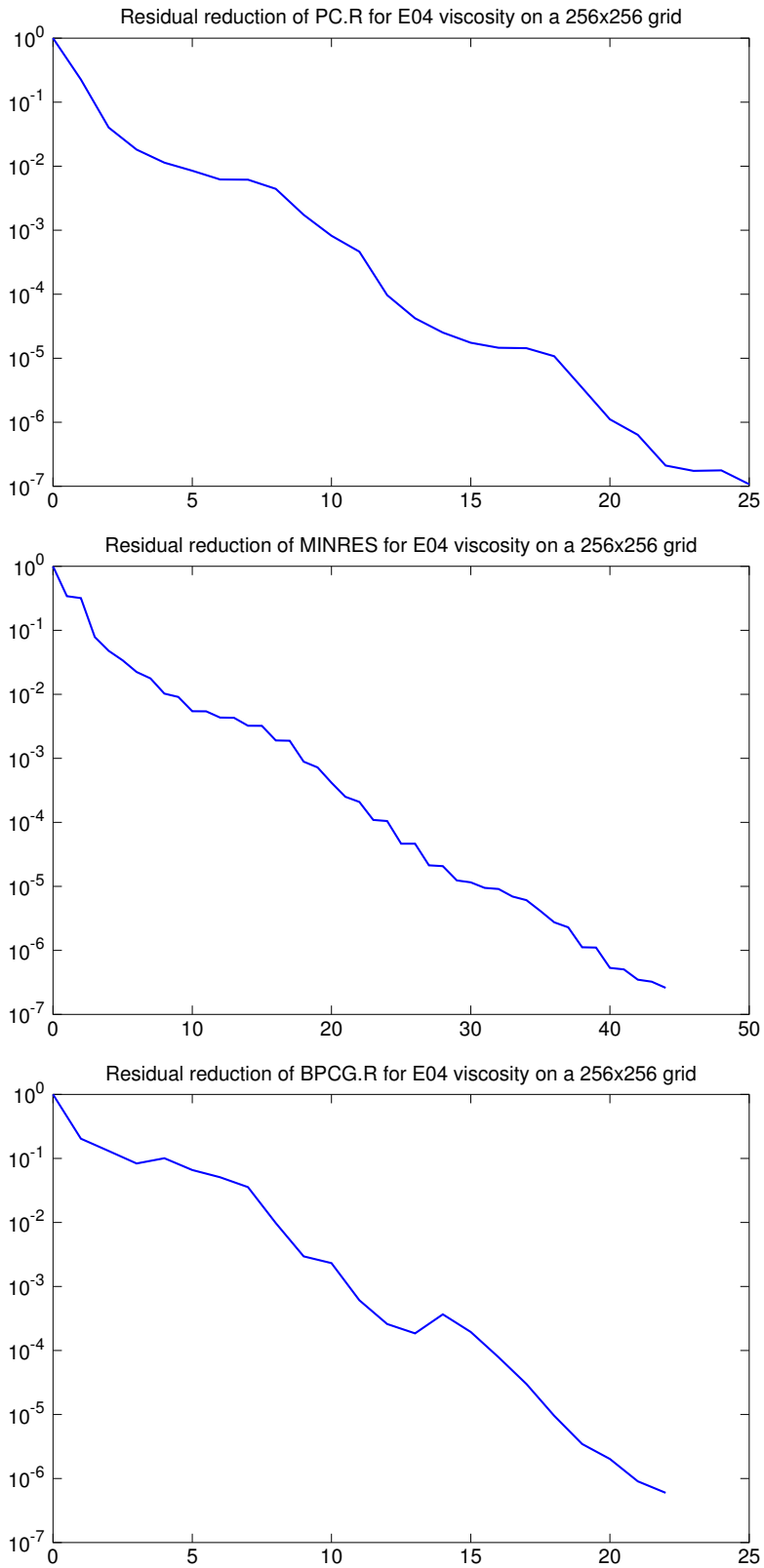

Figure 3.5: Residual reduction for E04 in Example 1

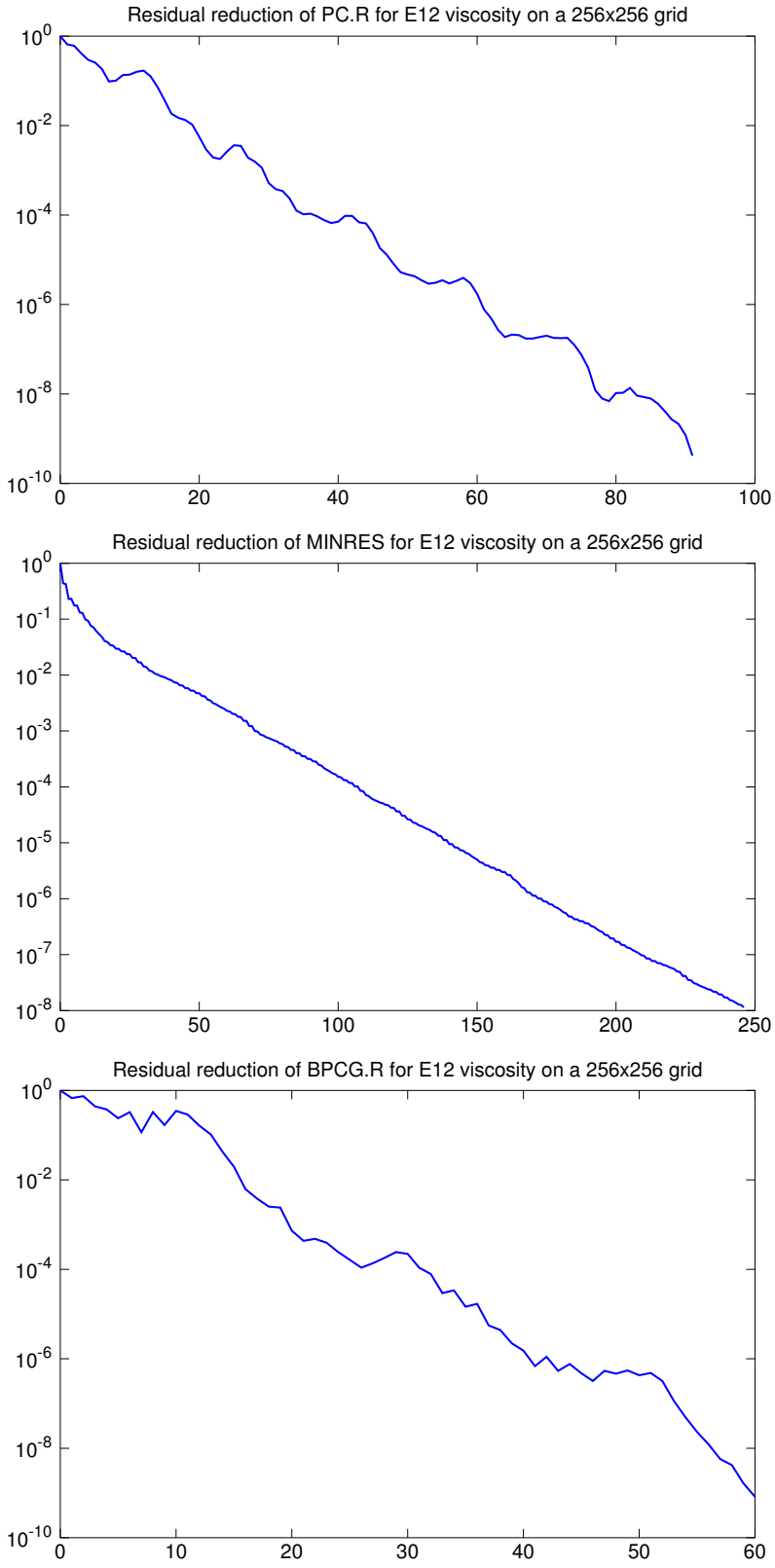

Figure 3.6: Residual reduction for E12 in Example 1

for I-structures. MINRES is also considered a parameter-free method, which is often seen as its main advantage. But if one considers the stopping criterion for the application of the preconditioner as a parameter, MINRES is not entirely parameter free. It has a stronger dependence on *itol* than PC.R and BPCG.R have, also restarting MINRES cannot alleviate this drawback. However, it is the only method in this study not requiring a restarted algorithm to run efficiently with variable viscosity. BPCG.R performs very close to PC.R, dropping back a little for I08 and I12. A disadvantage regarding the implementation is the additional factor k_{bp}, which has to ensure positive definiteness of $\mathbf{A} - \mathbf{A}_0$. With the worst-case estimate (3.37), BPCG.R was the slowest of the three methods, especially for the S-structures, where (3.37) leads to $0.5 \leq k_{bp} \leq 0.67$. The optimal value for every specific viscosity structure can only be found by experimentation or experience in parameter choices. Initial experiments with fixed values of $0.8 \leq k_{bp} \leq 0.98$ showed that it was mainly the inner accuracy which changed its optimal value to higher precision as k_{bp} approached 0.98, whereas it_i mostly stayed within a 15%-interval.

While several comparisons of Stokes solvers applied to finite-element discretizations have been published (Elman, 1996; Peters et al., 2005; Larin and Reusken, 2008; Geenen et al., 2009; ur Rehman, 2009), only the last two of them consider variable viscosity.

The results shown here confirm some findings of Elman (1996), who considered, in addition to other elements, a stabilized $Q_1 - Q_1$ finite-element discretization, even though with a penalty term. As he compared solvers similar to those in this study, he also found their performances being very close to each other. In detail, he found BPCG to be more efficient than the original method by Bramble and Pasciak (1988) and also more efficient than MINRES. In SSST, the BPCG is more efficient than MINRES only for E- and I-profiles and requires BPCG to be restarted. He also found the overall computational cost of MINRES to increase as the accuracy of the preconditioner is enhanced, which has also been confirmed in SSST. Compared to the findings of Peters et al. (2005), who also considered scaling of the Schur complement S by a factor ranging from 10^{-4} to 10^4, BPCG.R and MINRES perform well, again indicating that $\mathbf{A}$ and S are properly scaled. As they do not consider viscosity variations, they were able to run BPCG without restarts. PC.R and BPCG needed essentially the same it_i, while MINRES needs about 1.5 to 1.7 times that. This is exactly the same result as in Examples 1 and 2 for E04. This may be an artifact, because they use a three-dimensional Taylor-Hood $P_2 - P_1$ finite element pair. Larin and Reusken (2008), who used the same $P_2 - P_1$ element, found MINRES being more costly than PC by 1.3.

Geenen et al. (2009), who considered smooth viscosity variations as well as viscosity jumps, also found that preconditioning of S with the viscosity-scaled pressure mass matrix M_η is crucial for providing h- and $\Delta\eta$-independent convergence of a Krylov method. Furthermore, ur Rehman

(2009) compared PC.R with a GMRES method using a block triangular preconditioner and found PC.R becomes ever more advantageous as the viscosity contrast increases.

All these findings indicate that one can transfer results from isoviscous studies to a mildly varying (very smooth gradients, moderate amplitude) viscosity case, and that PC.R can be expected to be the most efficient method for high viscosity variations.

3.7 Conclusion

Although PC.R is the most efficient method, especially with high viscosity variations, the difference among all the solvers we considered do not appear large enough to justify switching from an already existing efficient implementation of a Krylov solver to another one. Considering the small to medium effort in implementation and minimal parameter choices, MINRES is a good choice, although one must select the inner accuracy carefully. Thus MINRES does not offer a significant advantage over PC.R, which is in any case the simplest of the three methods. BPCG.R requires the most effort to implement efficiently, because $itol$, k_{bp} and a proper restarting criterion ($bptol$) have to be chosen. Moreover, the condition for positive definiteness of $\mathbf{A} - \mathbf{A}_0$ has to be checked regularly.

From what we see in this study, PC.R is the best suited algorithm for a new implementation of a variable-viscosity Stokes solver, because it has no significant drawback, neither in computational efficiency nor in time and effort to implement.

Chapter 4

3D-spherical Discretization

This chapter provides an overview of the piecewise linear finite-element discretization of the 3D-spherical mantle convection code Terra and then describes its stabilization in detail. Discretization errors are estimated, and the effect of the stabilization operator is shown for different types of spurious pressure oscillations in comparison to the discrete gradient of these pressures. The maximum divergence error requires weighting of the stabilization matrix up to grid level 8 or 10, depending on the input data, thus reducing its effect on coarser grids. It is done adaptively during the convection calculation to get the highest possible weight, i.e., the strongest effect of stabilization, in every time step. Terra's variable viscosity formulation, together with various approaches to average nodal viscosities and their implementation to the stabilization matrix are described. Time discretization and the feasibility of using FEM-libraries are briefly discussed.

4.1 Computational Grid

Baumgardner (1983) used the projection of the regular icosahedron onto the sphere as a starting point in constructing an almost uniform triangular grid over the sphere. That coarse grid is then refined successively by dyadic subdivision, connecting the midpoints of the previous grid's edges, until the desired resolution is reached. A sequence of grids, up to grid level 5, the coarsest resolution used in this study, is shown in Figure 4.1. Pairs of icosahedral triangles may be combined to obtain ten logically rectangular diamond-shaped domains. Baumgardner and Frederickson (1985) define the basis functions on this grid by an iterative process using barycentric coordinates and provide estimates of the discretization error. Boal et al. (2008) confirm in theory the continuity of this process. They also prove the quasi-uniformity of the sequence of inscribed polygonal surfaces which converge to the spherical triangles. For the planar triangles, underlying our grid, they provide constants to estimate the remaining non-uniformity on the lowest 10 grid levels. These results are shown in Figure 4.2.

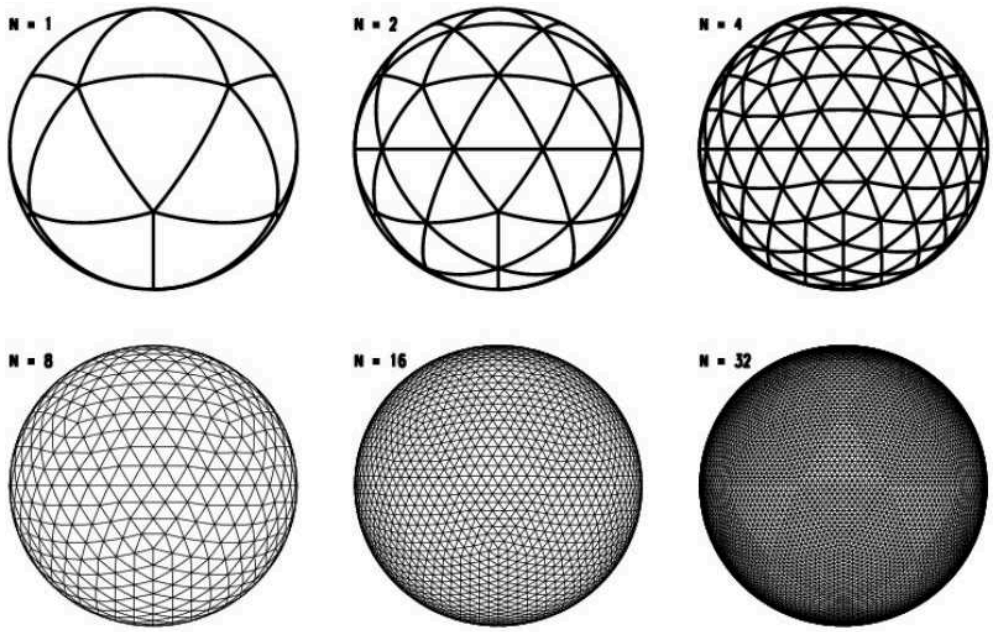

Figure 4.1: Dyadic icosahedral triangulations of the two-sphere from Baumgardner (1983)

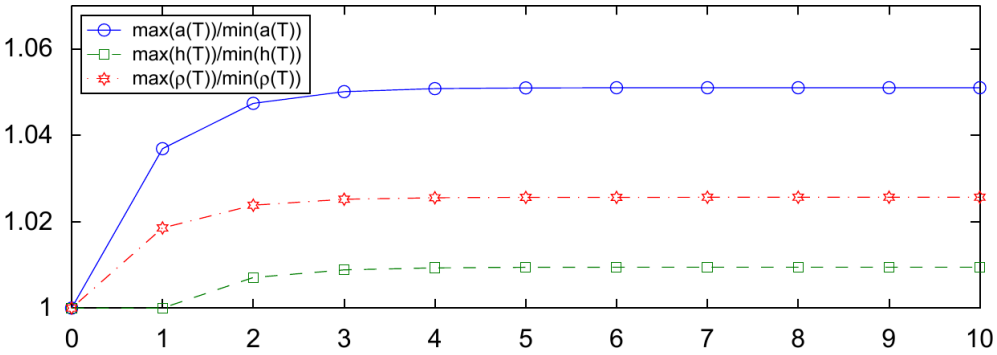

Figure 4.2: Quasi-uniformity of triangle area a, edge length h and inradius ρ for the 10 lowest grid levels of the icosahedral grid from Boal et al. (2008)

This 2D lateral grid is replicated radially within the spherical shell to obtain the 3D grid. The ratio of inner radius r_i to outer radius r_o is approximately 0.55. The number of radial layers is always a power of two, usually half the number of horizontal subdivisions along an icosahedral edge. Radial spacing in Terra can be chosen to be equidistant, stretched towards the boundaries or proportional to the radius by an input switch. Stretched spacing means that a cosine function is added to a constant function to yield narrower spacing near the boundaries and wider spacing in the middle of the shell. Proportional spacing leads to geometrically similar cells with radius with an aspect ratio close to one.

4.2 Finite-element Operators in the Spherical Shell

While early versions of Terra calculated the finite-element operators on a grid much finer than the grid used for the convection simulation, increased resolution now allows to calculate the operators simply on the grid where they are used. The calculation of the finite-element operators in Terra is described in detail by Baumgardner (1983) and Yang (1997). The latter describes the extension of $\mathbf{A}$ from a scalar Laplacian in each dimension to a variable-viscosity tensor operator. To continue these descriptions, we here adopt their naming conventions instead of those used in Chapter 2.

The three-dimensional basis functions N_i are decomposed into radial parts and two-dimensional tangential parts as follows:

$$N_i(\mathbf{r}) = M_i(r) \cdot L_i(\theta, \phi). \tag{4.1}$$

The piecewise linear lateral functions are defined by an iterative process on ever finer grids, they calculated on the sphere and can be visualized as shown in Figure 4.3. The volume integral over a three-dimensional basis

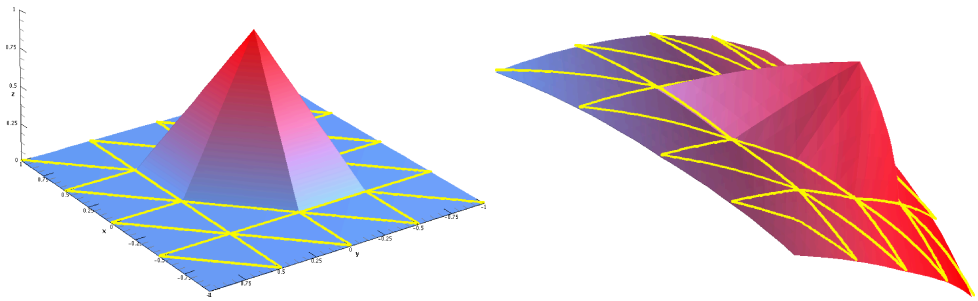

Figure 4.3: Bilinear lateral basis functions, taken from Müller (2008).

function is

$$\int_\Omega N_i(\mathbf{r}) dV = \int_{r_i}^{r_o} M_i(r) r^2 dr \cdot \int_S L_i(\theta, \phi) dA, \tag{4.2}$$

where S is the unit sphere. Note that the spherical volume element is $r^2 \sin\phi$. However, although covering the unit sphere, tangential integration uses Cartesian coordinates. Thus, $\sin\phi$ is omitted from the volume element. By exploiting this decomposition into radial and tangential parts, one may avoid storing the full volume integrals for the finite-element operators are not stored in Terra. The radial part needs to be computed only along a single radial line. Similarly, the tangential part needs to be computed over a small portion of the spherical surface because of the symmetries in the lateral grid. Because of the compact support of the basis functions, the integrals in (4.2) vanish everywhere outside the region of basis support. Figure 4.4 shows the volume where a representative basis function N_i assumes non-zero values. It implies that each local finite element operator is a 21-point stencil encompassing 12 adjacent triangular

prism cells. It also shows the lateral numbering of neighboring points and areas, adjacent to the center node. In computing the tangential operator

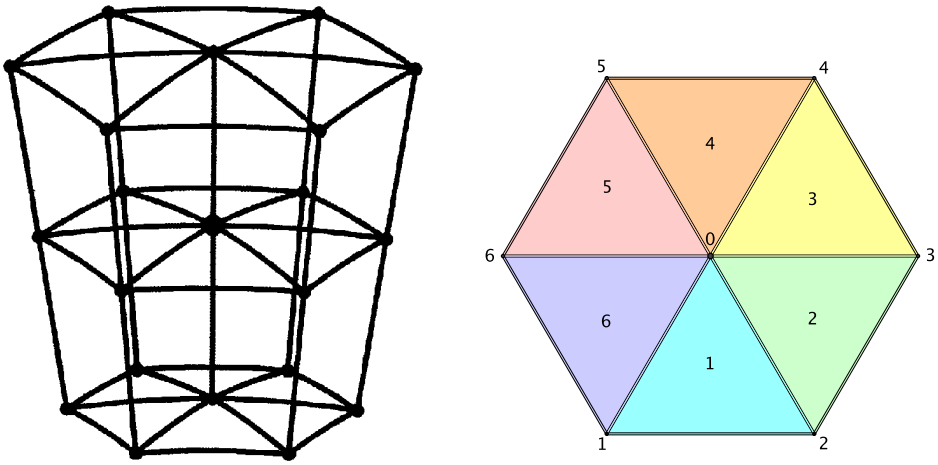

Figure 4.4: Left: A typical 21-point stencil, where a nodal basis function associated with the center node does not vanish. Taken from (Yang, 1997) Right: Lateral numbering of neighboring points and adjacent areas of the center node

terms, Terra iterates over all, usually six, adjacent areas and includes the appropriate integral if both basis functions do not vanish on that area. As an example, the tangential operator term, that connects the center point 0 to point 2, includes the integrals of the two basis functions over triangles 1 and 2. Except for the pentagonal nodes in the lateral grid which have only five neighbor nodes, seven lateral integrals are computed and stored, connecting the point to itself and to its six nearest neighbors. This is done for one diamond only, because the diamonds are congruent. Radial operator terms are computed along a single radial line. For every nodal layer three integrals are computed and stored. These integrals involve the radial basis function associated with that layer and the radial basis function associated with the layer above, that layer, and the layer below, respectively. Integrals over non-existent layers at the boundaries are set to 0. A large set of tangential and radial operator terms needed for the momentum, gradient and mass matrix operators are listed in (Baumgardner, 1983, Tables 4.2. and 4.4). However, not all of these are still in use in the current Terra code, because at that time, piecewise constant pressure functions were used. In the current version, not only pressure, but also temperature, density and viscosity are represented in terms of the nodal basis functions.

4.2.1 Discretization of Mass and Stabilization Matrices

The calculation of the stabilization matrix follows exactly the description in Section 2.5, and it is implemented using the decomposition (4.1). Let N_i denote the piecewise linear pressure basis function associated with point

i, and let K_i denote the projection of N_i onto the space of piecewise constant functions R_0, given in (2.28). As R_0 forces its elements to be constant only within a single grid cell, K_i need not be constant all over the 21-point stencil. It is defined element-wise as

$$\int_{\Omega_e} K_i = \int_{\Omega_e} N_i, \tag{4.3}$$

leading to

$$K_i = \frac{\int_{\Omega_e} N_i}{|\Omega_e|}. \tag{4.4}$$

Using (2.32), this leads to the following elemental contributions to the stabilization matrix C:

$$\begin{aligned} c_{e\_ij} &= \frac{1}{\eta_e}\left[\int_{\Omega_e} N_i N_j dV - \int_{\Omega_e} K_i K_j dV\right] \\ &= \frac{1}{\eta_e}\left[\int_{\Omega_e} N_i N_j dV - \int_{\Omega_e} \frac{\int_{\Omega_e} N_i dV}{|\Omega_e|}\frac{\int_{\Omega_e} N_j dV}{|\Omega_e|} dV\right] \\ &= \frac{1}{\eta_e}\left[\int_{\Omega_e} N_i N_j dV - \frac{\int_{\Omega_e} N_i dV \int_{\Omega_e} N_j dV}{|\Omega_e|}\right]. \end{aligned} \tag{4.5}$$

As the first integral in (4.5) is the elemental contribution to the mass matrix, the mass matrix is computed and assembled as a by-product of the stabilization matrix. For these two matrices, four types of elemental integrals must be computed:

$$\begin{aligned} \eta_e m_{e\_ij} &= \int_{\Omega_e} N_i N_j \\ &= \int_{r_b}^{r_t} M_i M_j r^2 dr \int_{\triangle_e} L_i L_j dA \end{aligned} \tag{4.6}$$

$$\begin{aligned} \eta_e p_{e\_ij} &= \frac{\int_{\Omega_e} N_i dV \int_{\Omega_e} N_j dV}{|\Omega_e|} \\ &= \frac{\int_{r_b}^{r_t} M_i r^2 dr \int_{r_b}^{r_t} M_j r^2 dr}{(r_t^3 - r_b^3)/3} \frac{\int_{\triangle_e} L_i dA \int_{\triangle_e} L_j dA}{|\triangle_e|} \end{aligned} \tag{4.7}$$

where $\triangle_e$ is the triangle on the unit sphere, and $\{r_t, r_b\}$ are the top and bottom radii corresponding to Ω_e. The basic integrals are:

$$\int_{r_b}^{r_t} M_i M_j r^2 dr = \begin{cases} \frac{\frac{1}{5}(r_t^5 - r_b^5) - \frac{1}{4}(r_b + r_b)(r_t^4 - r_b^4) + \frac{1}{3} r_b r_b (r_t^3 - r_b^3)}{(r_t - r_b)^2} & i, j \text{ on top layer} \\ \frac{\frac{1}{5}(r_t^5 - r_b^5) + \frac{1}{4}(r_t + r_b)(r_t^4 - r_b^4) - \frac{1}{3} r_t r_b (r_t^3 - r_b^3)}{(r_t - r_b)^2} & i, j \text{ on top and bottom} \\ \frac{\frac{1}{5}(r_t^5 - r_b^5) - \frac{1}{4}(r_t + r_t)(r_t^4 - r_b^4) + \frac{1}{3} r_t r_t (r_t^3 - r_b^3)}{(r_t - r_b)^2} & i, j \text{ on bottom layer} \end{cases} \tag{4.8}$$

$$\int_{r_b}^{r_t} M_i r^2 dr = \begin{cases} (r_t - r_b)(\frac{1}{4}r_t^2 + \frac{1}{6}r_t r_b + \frac{1}{12}r_b^2) & i \text{ on top layer} \\ (r_t - r_b)(\frac{1}{12}r_t^2 + \frac{1}{6}r_t r_b + \frac{1}{4}r_b^2) & i \text{ on bottom layer} \end{cases} \tag{4.9}$$

$$\int_{\triangle_e} L_i L_j dA = (1 + \delta_{ij}) \frac{|\triangle_e|}{12} \tag{4.10}$$

$$\int_{\triangle_e} L_i dA = \frac{|\triangle_e|}{9} \tag{4.11}$$

As the volume element depends on the radius, the radial integrals depend on the basis function's location within the layer. Here δ_{ij} is the Kronecker delta of the positions of i and j on the unit sphere.

4.2.2 Properties of the Stabilization Matrix

The main purpose of the stabilization matrix is to complement the gradient matrix. It should add significant terms to the mass equation when the discrete pressure gradient is zero but when the continuous gradient is not. But it should not add significant terms when the discrete pressure gradient is close to the continuous one. What makes evaluation of the stabilization difficult is that a pressure function with non-vanishing continuous gradient but vanishing discrete gradient cannot be found easily on the spherical grid. Even a pure radial pressure oscillation has a non-vanishing discrete gradient. Thus it is instructive also to show the discrete gradient for comparison when the stabilization matrix is applied to a specified pressure field.

Table 4.1 shows norms of the stabilization term and of the discrete gradient for constant, linear varying, radial oscillating and checkerboard pressure fields. It also shows the ratio of stabilization term to discrete gradient. This is expected to decrease by a factor of $\sqrt{8}$ with every grid level as the stabilization matrix contains a factor of volume, whereas the gradient matrix contains a factor of the volume's square root. For the linear varying pressure field L, we get exactly that factor. For the oscillations we get only $\sqrt{2}$. This indicates that the stabilization term emphasizes spurious pressures more strongly by a factor of 2 with every grid refinement. It is worthy of note, that the results in Table 4.1 are independent of the oscillation's magnitude down to machine precision.

The stabilization uses projections of the pressure into the piecewise constant space. Thus pressure accuracy of a stabilized Q1-Q1 discretization is expected to be of one degree less than the accuracy of a piecewise linear pressure. However, Dohrmann and Bochev (2004) and Elman et al. (2005) found that it is between first and second order, both, in two and three dimensions. The loss in pressure accuracy, compared to a Q2-Q1 stable element is only of order $1.2 - 1.5$.

Table 4.1: Norms of stabilization term and discrete gradient for constant (C), linear (L), radial oscillating (R-O), tangential oscillating (T-O) and 3-D oscillating (3D-O) pressures on different grid levels in a spherical shell with $r_i = 0.55$, $r_o = 1$, $p = 1$, $\Delta p = 0.3$.

l	C	L	R-O	T-O	3D-O
$\|\|C * p\|\|$					
5	6.02×10^{-18}	2.18×10^{-7}	1.07×10^{-6}	7.13×10^{-7}	1.48×10^{-6}
6	2.70×10^{-18}	1.96×10^{-8}	1.37×10^{-7}	9.14×10^{-8}	1.90×10^{-7}
7	2.17×10^{-18}	1.75×10^{-9}	1.74×10^{-8}	1.16×10^{-8}	2.40×10^{-8}
8	6.77×10^{-19}	1.55×10^{-10}	2.18×10^{-9}	1.46×10^{-9}	3.02×10^{-9}
$\|\|B^T * p\|\|$					
5	4.35×10^{-20}	1.68×10^{-4}	4.35×10^{-5}	9.94×10^{-6}	5.14×10^{-5}
6	1.18×10^{-20}	4.40×10^{-5}	7.92×10^{-6}	1.80×10^{-6}	9.35×10^{-6}
7	2.97×10^{-21}	1.12×10^{-5}	1.42×10^{-6}	3.25×10^{-7}	1.67×10^{-6}
8	7.48×10^{-22}	2.84×10^{-6}	2.53×10^{-7}	5.78×10^{-8}	2.99×10^{-7}
$\|\|C * p\|\|/\|\|B^T * p\|\|$					
5	1.39×10^{2}	1.30×10^{-3}	2.46×10^{-2}	7.17×10^{-2}	2.88×10^{-2}
6	2.29×10^{2}	4.46×10^{-4}	1.73×10^{-2}	5.07×10^{-2}	2.03×10^{-2}
7	7.31×10^{2}	1.56×10^{-4}	1.22×10^{-2}	3.56×10^{-2}	1.43×10^{-2}
8	9.05×10^{2}	5.47×10^{-5}	8.62×10^{-3}	2.52×10^{-2}	1.01×10^{-2}

4.2.3 Weighting of the Stabilization Matrix

A significant drawback, mentioned briefly in (Dohrmann and Bochev, 2004), is that the stabilized formulation can lead to an increase of the maximum divergence error compared to its stable counterpart. This is observed also in our spherical implementation. As a remedy they propose to weight the whole stabilization matrix with a constant factor α. However, neither a detailed analysis nor an effective method for choosing α are given.

The problem in our case is that Cp in (3.7) is often so high that it already equals out $B\mathbf{A}^{-1}B^Tp$ before it reaches its specified upper threshold. From that two question arise:

- Can α be set small enough to allow $B\mathbf{A}^{-1}B^Tp$ to be reduced below its threshold while still providing sufficient stabilization?
- Is the accuracy of $B\mathbf{A}^{-1}B^Tp$ with $\alpha = 1$ already sufficient?

In many cases neither of the two questions can be answered in the affirmative. Therefore, the selection of α is done adaptively. After choosing an appropriate residual reduction rate for (3.7), the initial value of α will be decreased step-wise if reducing the residual of (3.7) does not lead to $B\mathbf{A}^{-1}B^Tp$ being sufficiently small. Because such a low α will not be required in later time steps, it is increased later when $B\mathbf{A}^{-1}f$ gets larger, as long as $B\mathbf{A}^{-1}B^Tp$ is maintained below the threshold. As the detailed parameter settings for computing α depend not only on relative residuals

but also on iteration counts, we postpone discussion of these issues until Chapter 5.

4.2.4 Effect of Stabilization to the Discretization

It should be mentioned that prior to the implementation of the stabilization matrix, Terra was typically run with a nonzero bulk viscosity bv, added in the formulation of the stress tensor in (1.5); i.e., the the factor $\frac{1}{3}$ multiplying $\delta_{lm}\frac{\partial u_k}{\partial x_k}$ was increased effectively to $\frac{2}{3}$ or even to one. An important advantage of using the stabilization matrix is that we can now get rid of the nonzero bulk viscosity which leads to a penalized **A** operator, that produces slightly incorrect velocity solutions. This error, though not large, is evident in results from benchmark convection cases. As an example, results from a standard 3-D spherical convection case for a steady-state tetrahedral (L = 3, m = 2) pattern with constant viscosity and Rayleigh number of 7000 are displayed in Table 4.2. The Nusselt number Nu, a diagnostic of convective vigor, displays clear dependence on bv. The unstabilized $bv = 0$ and stabilized results, however, are in close agreement.

Table 4.2: Nusselt numbers for the Ra=7000 steady-state tetrahedral (L = 3, m = 2) flow pattern with constant viscosity for different settings of bulk viscosity bv (columns 2-4) and for a stabilized formulation (column 5) for the Earth's mantle ($r_o/r_i = 0.5463$). In column 6, the radii ratio is 0.55 and $bv = 0$. A 10x65x65x33 grid is used.

Formulation	$bv = 2/3$	$bv = 1/3$	$bv = 0$	stab	$r_i = 0.55r_o$
Nu	3.481	3.506	3.534	3.543	3.514

4.3 Variable Viscosity

Terra offers several viscosity formulation options. This section does not give a thorough description of them but merely summarizes those used in the selected examples cases in this dissertation.

A quite simple method, V2, utilizes a specified radial viscosity profile and a lateral variation, that depends exponentially on temperature as follows

$$\Delta\eta = \Delta\eta_r(r)e^{\beta(T_{ref}-T)}, \tag{4.12}$$

where the factor β controls the strength of the lateral variation. With $\beta = 0$, this method reverts to V1, involving a radial viscosity profile only.

Another commonly used method, V3, applies the Arrhenius law

$$\Delta\eta = e^{\frac{E+pV}{RT}}, \tag{4.13}$$

where E is the activation energy and V is the activation volume. This is the standard method for modeling a primarily temperature-dependent

viscosity. To account for compositional effects, radial pre-factors are also available as options. In each of these methods, the user can specify minimum and maximum values for $\Delta\eta$ as for η itself.

The method, tailored most specifically to the Earth's mantle, V4, is an extension of the formulation of Walzer et al. (2004b). It uses the formulation

$$\eta = 10^{rn} \exp\left\{ c \left(\frac{\overline{T_m}}{\overline{T}} - \frac{\overline{T_m}}{\overline{T_{st}}} \right) \right\} \eta_r(r) \exp\left\{ ct \left(\frac{T_m(r)}{T(r,\theta,\phi)} - \frac{T_m(r)}{T_{av}(r)} \right) \right\}, \tag{4.14}$$

with a precomputed radial viscosity profile η_r and precomputed laterally averaged melting temperatures T_m, based on mantle mineralogy, thermodynamics, seismology and postglacial rebound. During the mantle's convection and evolution, the radial profile is shifted when the globally averaged temperature $\overline{T}$ deviates from the globally averaged starting temperature $\overline{T_{st}}$. It can also be shifted independent of space and time by specifying the parameter rn. The lateral viscosity variation is normalized by comparing the local temperature to the laterally averaged temperature each time viscosity is updated.

If desired, viscosity variations may be updated each time step. However, since changes between successive time steps tend to be relatively small, they are typically updated only every 5 time steps (see Section 4.4).

4.3.1 Viscosity Averaging in the Operators

Currently in Terra the computation of the radial and tangential operator components is done without any viscosity dependence, i.e. in (2.22) $\eta_k \psi_k$ is set equal to 1. Viscosity variation is folded in only when the operator is applied. Because integration has already been done, only a constant factor can be multiplied to the operator parts. Therefore, it is necessary to average nodal viscosities to cell-wise constant values. Moreover, as the basic integrals are summed over the whole overlapping area of two basis functions, viscosity must be averaged within a volume larger than a grid cell. This volume can span as many as twelve cells when $i = j$, i.e., a point is connected with itself (see Figure 4.4). It can also span only two cells when i, j are adjacent in both, radial and tangential, dimensions.

When averaging viscosity, one has to decide which nodal values should be taken into account and how they are averaged. Yang (1997) used the geometric mean of the two nodal viscosities where the basis functions are one:

$$\eta_{ij} = \sqrt{\eta_i \eta_j}. \tag{4.15}$$

When $i = j$, only the viscosity of the center node is used as an average for the whole 21-point stencil, containing 12 cells. While the geometric mean yields the arithmetic mean of the logarithms of the viscosities, which is a reasonable choice, taking the average of only one or two points is not

desirable. It is desirable to have cell averaged viscosities from all surrounding nodal values. The computation of such cell averaged viscosities had already been included in Terra, mainly to switch from a nodal-based to a cell-based representation, and is now used again. Based on the findings of Deubelbeiss and Kaus (2008), who evaluated the accuracy of different viscosity interpolation methods for Stokes flow, I chose to use the harmonic mean of the n nodal values here:

$$\eta_e = \frac{n}{\frac{1}{\eta_1} \cdots + \ldots \frac{1}{\eta_n}} \tag{4.16}$$

This was shown to be the most accurate one, with geometric mean being a bit worse and arithmetic mean being much worse. The harmonic mean emphasizes the low viscosities which is physically appropriate, as the main flow occurs in the low viscosity region of the cell when viscosity varies strongly.

When assembling the stabilization matrix, also the cell viscosities are averaged harmonically. Within a radial layer this is sufficiently accurate, because the triangle areas differ by at most 5% globally (see Figure 4.2). The difference between harmonic and geometric mean would be much higher in the presence of strong variations. But if the overlapping volume of two basis functions spans two radial layers, the cell viscosities are weighted by the volume fraction of their radial layer before being averaged, which is very accurate. Because the mass and stabilization matrices also contain a factor of volume, this volume weighting yields almost exactly the same as if the basic integrals would already be computed using cell viscosities. The following calculation shows this equality in case that the cell integrals are exactly proportional to the cell volume:

V_1, V_2 cell volumes

η_1, η_2 cell viscosities

$\eta_{mean} = \dfrac{V_1 + V_2}{\frac{V_1}{\eta_1} + \frac{V_2}{\eta_2}}$ volume-weighted harmonic mean

$$m_1 = k\frac{V_1}{\eta_1} + k\frac{V_2}{\eta_2} \quad \text{local matrix} \tag{4.17}$$

$$m_2 = k\frac{V_1 + V_2}{\eta_{mean}} = k(V_1 + V_2)\frac{\frac{V_1}{\eta_1} + \frac{V_2}{\eta_2}}{V_1 + V_2} = k\frac{V_1}{\eta_1} + k\frac{V_2}{\eta_2} \tag{4.18}$$

$$m_1 = m_2 \quad \text{q.e.d.} \tag{4.19}$$

Whereas the radial operator parts use slightly different values of k in (4.17), the tangential parts exactly fulfill (4.17). Then also (4.19) holds exactly, thus it is also possible to get rid of the small irregularities in tangential viscosity averaging if necessary. Note that the derivation of (4.19) requires viscosity in the mass and stabilization matrices to be averaged harmonically.

As the mass matrix M serves as a preconditioner for the Schur complement $S = B\mathbf{A}^{-1}B^T + C$, it should resemble the viscosity averaging of $\mathbf{A}^{-1}$. So the choice of a simple 2-point geometric mean, the current Terra code uses for assembling $\mathbf{A}$, is also provided for M and C. But this usually leads to a higher norm of the stabilization term and increases the maximal divergence errors (see Section 4.2.3). Then the weighting factor α for the stabilization matrix has to be decreased.

4.4 Time Discretization

Terra performs explicit time stepping using a second-order Runge-Kutta scheme. The maximal step size is determined by the Courant-criterion, that is, the flow field must not pass more than a whole grid cell in one step. Every time step is a computed as a fraction δ of the Courant step. The fraction δ depends on iteration numbers and residual reductions of the Stokes and velocity solvers and is limited from above by a limit, set by the user. As the computation of δ has been changed in this work, it is described in Chapter 5.

4.5 Remarks on the Discretization

The above-mentioned computations of α and δ are not the only occurrences in Terra, where the discretization gets feedback from the solver and adapts accordingly to get robust convergence. From the programming point of view, such feedbacks significantly enlarge the interface between discretization and solver from a uni-directional to a bi-directional one. When using libraries to set up the discretization, this would require them to be highly tailored to the convergence requirements and the solver's feedback as well as to the complex physical model in Terra.

Chapter 5

3D-spherical Stokes Solver

This chapter describes the implementation of scaling and pressure correction, described in Chapter 3, in the Terra code. In contrast to SSST, the velocity solution is obtained using a multigrid algorithm with matrix-dependent transfer operators. This algorithm is described in detail in Yang and Baumgardner (2000) for the 2D-Cartesian version and in Yang (1997) for the 3D-spherical version of the Terra code. In 3D-spherical with variable viscosity, it does not perform as well as expected from the 2D results in Yang and Baumgardner (2000) (see also (Tackley, 2008)). Regarding the pressure correction algorithm, however, stopping tolerances and restart criteria are chosen quite similar to those in SSST, although in some cases the matrix-dependent transfer multigrid puts a limitation to the attainable accuracy of the inner solver. As in Chapter 3, the convergence behavior of outer and inner solver is examined for different viscosity settings.

5.1 Example Problems

Four example problems are used to examine the convergence behavior of the Stokes solver. The radial temperature and viscosity profiles of these, together with the maxima of their lateral variations, are shown in Figures 5.1 and 5.2 for the beginning and for advanced convection.

The first two examples are benchmark problems with a steady-state solution, which are commonly used in the mantle-convection community. The first, Case 002, which has already been discussed in Section 4.2.4, is the standard 3-D spherical case for a tetrahedral (L = 3, m = 2) convection pattern with constant viscosity and a Rayleigh number of 7000. The second, Case 007, differs from Case 002 only in having a slightly varying temperature-dependent viscosity with $\Delta\eta = 20$. It uses viscosity formulation V2 (see Section 4.3). However, as seen in Figure 5.2, the temperature variation after reaching steady state is so high that almost all of the viscosity variation occurs laterally, except at the boundaries. Results of various authors have been summarized by Stemmer et al. (2006) and Zhong et al.

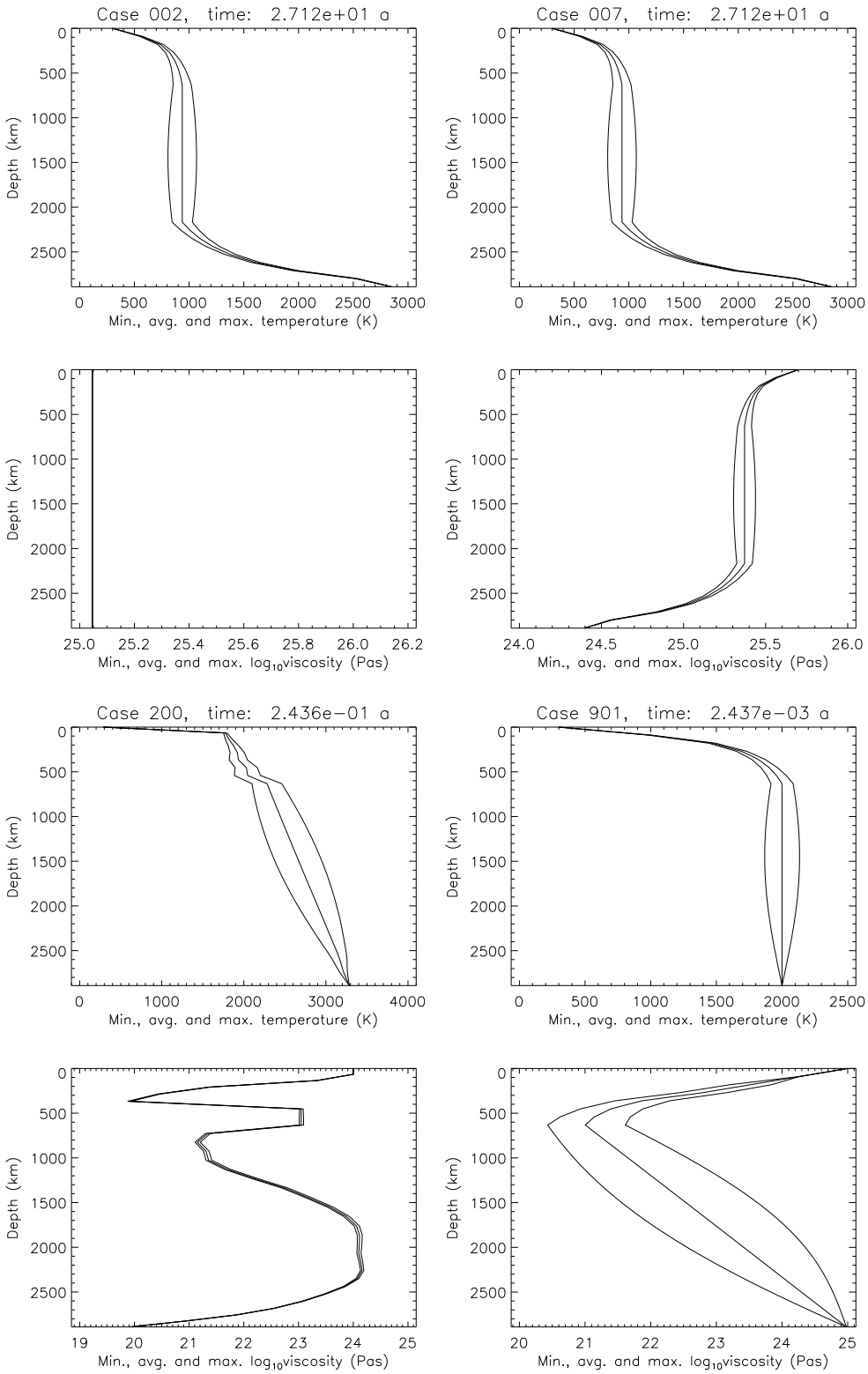

Figure 5.1: Minimal, averaged and maximal temperatures and viscosities as a function of depth for the example cases 002, 007, 200 and 901 at the beginning of the convection calculation.

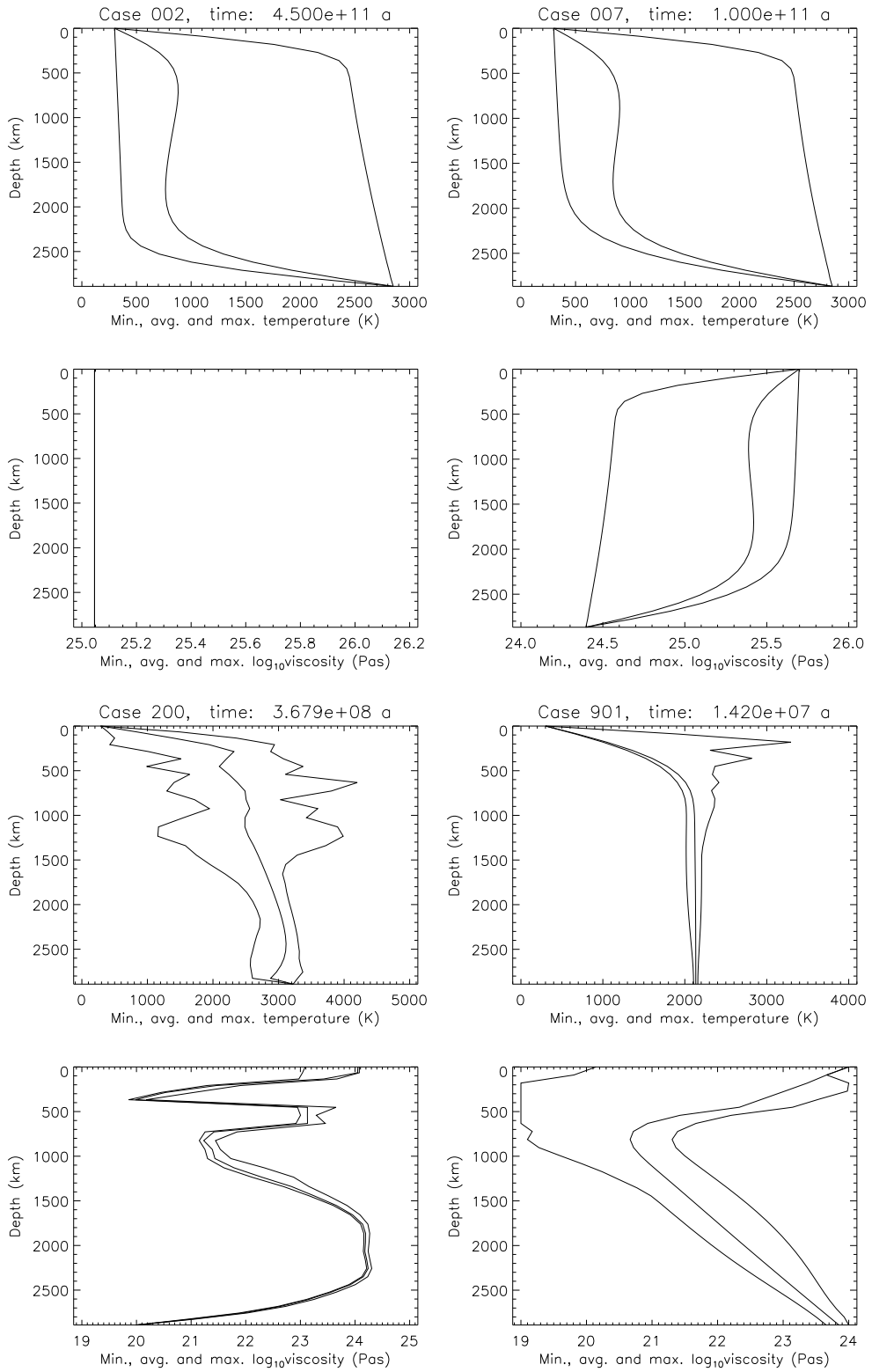

Figure 5.2: Same as Figure 5.1, but after reaching steady-state (Cases 002 and 007) or running 5000 time steps (Cases 200 and 901). The temperature variation has strongly increased in all cases.

(2008) for these cases. To validate the Terra code, Richards et al. (2001) also used Case 002, but with $\frac{r_i}{r_o} = 0.5$.

The third example, Case 200, uses the viscosity formulation V4, together with the radial profile of Walzer et al. (2004b). As can be seen in Figure 5.1, the lateral viscosity variation of Case 200 is strongly damped, so that the numerical challenge lies in the strong radial variation together with the steep gradients. In its evolution, this case shows strong lateral temperature variations with high and low temperature peaks in the stiff transition layer and in the soft layers beside, respectively. Case 200 also uses internal and bottom heating, with internal heating decreasing over time, and the radial discretization uses slightly stretched spacing. The global shifting parameter rn in (4.14) is set to -0.65, the lowest value used in a simulation up to now.

The fourth example, Case 901, includes lateral variations in full magnitude, allowing viscosity to vary from 10^{19} to 10^{25} Pas. It also uses a radial profile which alone spans almost four orders of magnitude, although not with gradients as steep as in Case 200. As far as I know, this case has never been used in a convection simulation before, because of numerical instabilities.

5.2 Scaling and Preconditioning

The diagonal scaling (3.2) had already been implemented in Terra for the velocity with $x_{ii} = 1/\sqrt{\Delta\eta_i}$. $\Delta\eta_i$ denotes the viscosity variation from the global mean viscosity $\bar{\eta}$, and scaling is applied after the momentum equation had been divided by $\bar{\eta}$. Because of the 2-point geometric mean (4.15), used to assemble $\mathbf{A}$, we have $\Delta\eta_i = a_{ii}$. Prior to this work, the Schur complement S had not been scaled, instead the constant viscosity mass matrix M was used as a preconditioner for S. The system to solve was:

$$\begin{bmatrix} \mathbf{X} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{A} & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} \mathbf{X} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{X}^{-1} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{X} & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \mathbf{f} \\ g \end{bmatrix} \tag{5.1}$$

The inverse of M was approximated by 3-5 Jacobi iterations. When switching to M_η, however, such a quick inversion is not feasible.

In this work, I added $y_{ii} = 1/m_{jj}$, with m_{jj}, defined in (4.6), (4.8) and (4.10), with viscosity averaging described in Section 4.3.1. So we have exactly the scaled system (3.2), which is:

$$\begin{bmatrix} \mathbf{X} & 0 \\ 0 & Y \end{bmatrix} \begin{bmatrix} \mathbf{A} & B^T \\ B & -C \end{bmatrix} \begin{bmatrix} \mathbf{X} & 0 \\ 0 & Y \end{bmatrix} \begin{bmatrix} \mathbf{X}^{-1} & 0 \\ 0 & Y^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ p \end{bmatrix} = \begin{bmatrix} \mathbf{X} & 0 \\ 0 & Y \end{bmatrix} \begin{bmatrix} \mathbf{f} \\ g \end{bmatrix} \tag{5.2}$$

As in Section 3.1, in the following the subscripts are omitted from the scaled quantities for convenience of notation. Unscaled quantities are denoted with a tilde as $\tilde{\mathbf{u}}$, $\tilde{g}$, etc. Using scaled matrices is superior to using the mass matrix preconditioner, applied to unscaled matrices, which could be confirmed in our test cases.

5.3 Pressure Correction Algorithm

The use of the pressure correction algorithm as Stokes solver in Terra has been first mentioned by Yang (1997). With this work, it is changed from the standard mode (Algorithm 1 in Section 3.2), to a modified version of the restarted mode (Algorithm 2), which is shown as Algorithm 6. It also evaluates the current time step count *ncall* to decide when the right hand side of (3.7) has to be recomputed to give a rule for measuring residual reduction. As this changes only slightly with every time step, stopping tolerances do not lack precision when it is recomputed only every 50 or 100 time steps. Note that Algorithm 6 is slightly simplified compared to the actual implementation, but it does contain all conditional branches as well as most of the current parameter choices. An important aspect, resembling the considerations on error norms in Section 3.2.3, is that we do not want only the scaled divergence of velocity but also the unscaled divergence to be below a specified threshold. Therefore, extra checks on the reduction $\tilde{B}\tilde{\mathbf{u}}$ are performed. To retain the computational efficiency of Terra, scaling is done not earlier than within the execution of the pressure correction algorithm. Thus, the unscaled quantities are readily available.

The modification of the stabilization weight, here denoted γ, within the solver, clearly shows the feedback from solution to discretization. The user specifies a maximum weight as an input value, and within the pressure correction γ is adapted to a value as high as possible to get the divergence error below the specified threshold. As the convection evolves, γ usually rises by more than a factor of three from the value of the first time step, so that having a fixed value throughout the whole convection run would lead to a non-optimal choice. Therefore, γ is adapted to stay in a rather narrow, close to optimal interval.

5.4 Stopping Criteria and Solver Restart

As we have no eigenvalue estimates in the 3-D spherical model, choosing reasonable stopping criteria is usually subject to experimentation. The stopping tolerance *utol*, in Terra previously named *convtol*, for the multigrid algorithm to solve for $\mathbf{u}$ has been an input parameter from the very beginning of the Terra code. Reasonable values for *utol* range from 10^{-3} to 10^{-2}. As in SSST, a quantity controlling the accuracy of the inner iterations can be specified. It is named $vtolfac$ as it relaxes *utol* in the computation of the velocity search directions. The stopping tolerance for the Schur complement residual is also computed as a fraction of *utol*. It would be desirable to derive this fraction from mesh and model parameters but due to missing eigenvalue estimates it is implemented as input parameter $ptolfac$ in the current Terra version. When the unscaled divergence does not satisfy its tolerance, $ptolfac$ may be reduced to force higher accuracy in solving (3.7).

Algorithm 6: Pressure correction algorithm in Terra

if *mod(ncall,50) = 1* **then** Set rule for Schur complement residual.
  Solve $\mathbf{A}\mathbf{v} = \mathbf{f}$ for $\mathbf{v}$ until $\frac{||\mathbf{f}-\mathbf{A}\mathbf{v}||}{||\mathbf{f}||} < 10 \cdot utol$
  Compute $g = B\mathbf{v}$; $\tilde{g} = Y^{-1}g$; Reset $ptolfac$
end
if *ncall = 1* **then**
  $p_0 = 0$
else
  $p_0 = p$ from previous time step.
end
for *j=1* **to** N_o **do**
  Solve $\mathbf{A}\mathbf{u}_0 = \mathbf{f} - B^T p_0$ for $\mathbf{u}_0$ until $\frac{||\mathbf{f}-\mathbf{A}\mathbf{u}_0-B^T p_0||}{||\mathbf{f}||} < utol$
  Compute residual $r_0 = B\mathbf{u}_0 - \gamma C p_0$
  if $\frac{||\tilde{r}_0||}{||\tilde{g}||} < 0.8\, ptolfac \cdot utol$ **then** $\gamma = \min(1.1\,\gamma; \gamma_{max})$
  if $\left(\frac{||r_0||}{||g||} < ptolfac \cdot utol \ \wedge \ \frac{||\tilde{r}_0||}{||\tilde{g}||} < ptolfac \cdot utol\right)$ **then** Exit loop
  $stol = 1 - 2.4/(j+2)$
  for *i=1* **to** N_i **do**
    if *i=1* **then**
      $s_1 = r_0$
    else
      $\delta = \frac{\langle r_{i-1}, r_{i-1}\rangle}{\langle r_{i-2}, r_{i-2}\rangle}$
      $s_i = r_{i-1} + \delta s_{i-1}$
    end
    Solve $\mathbf{A}\mathbf{v}_i = B^T s_i$ for $\mathbf{v}_i$ until $\frac{||\mathbf{A}\mathbf{v}_i - B^T s_i||}{||B^T s_i||} < vtolfac \cdot utol$
    $\alpha = \frac{\langle r_{i-1}, r_{i-1}\rangle}{\langle s_i, B\mathbf{v}_i + \gamma C s_i\rangle}$
    $p_i = p_{i-1} + \alpha s_i$
    $\mathbf{u}_i = \mathbf{u}_{i-1} - \alpha \mathbf{v}_i$
    $r_i = r_{i-1} - \alpha(B\mathbf{v}_i + \gamma C s_i)$
    if $\frac{||r_i||}{||g||} < ptolfac \cdot utol$ **then** Exit loop
    if $\left(i > 1 \ \wedge \ \frac{||r_i||}{||r_0||} < stol\right)$ **then** Exit loop
  end
  $\mathbf{u}_0 = \mathbf{u}_i$; $p_0 = p_i$
  if $\frac{||r_i||}{||g||} < ptolfac \cdot utol$ **then** Watch also unscaled velocity divergence.
    if $\frac{||\tilde{r}_i||}{||\tilde{g}||} < ptolfac \cdot utol$ **then**
      Exit loop
    else
      if $j > 1 \ \wedge \ ||\tilde{B}\tilde{\mathbf{u}}|| < 1.05\, ||\gamma \tilde{C}\tilde{p}||$ **then**
        $\gamma = 0.9\,\gamma$
      else
        $ptolfac = 0.95\, ptolfac$
      end
    end
  end
end

As in SSST, the pressure correction is restarted after an intermediate residual reduction *stol* is reached. However, in Terra *stol* is not chosen to be constant. The reason is that also restarting cannot prevent the residual reduction curve from flattening with increasing iteration count. To accommodate to the convergence behavior, *stol* is given as a function of the outer loop count j by

$$stol = 1 - \frac{2.4}{j+2}, \tag{5.3}$$

leading to $stol = 0.2$ in the first and to $stol = 0.7$ in the sixth outer iteration. This is very close to the choice of Peters et al. (2005), who used $stol \in [0.2, 0.6]$. In addition to applying *stol*, the number of pressure correction iterations i per outer iteration j is confined to $i \in (2, 10)$ unless all residuals are within their specified thresholds.

For the four example cases, the default stopping criteria are given in Table 5.1. On grid level 6, these are used unless other choices are explicitly mentioned. These default values show that the cases with strongly varying

Table 5.1: Default Stopping criteria for the four example cases on grid level 6.

Case	002	007	200	901
$utol$	0.005	0.005	0.005	0.001
$vtolfac$	3.0	3.0	2.0	2.0
$ptolfac$	0.2	0.2	1.0	1.0

viscosity need a tighter *utol*. This would also be required for Case 200, but choosing a lower *utol* is not feasible because of the slow convergence of multigrid in that case. However, it indicates that the scaling with the square root of nodal viscosities cannot filter out all of the viscosity variations from the spectrum of the scaled operator (see (3.2)).

5.5 Time Stepping

The fraction δ, controlling how much of a grid cell the flow field is allowed to pass in a single time step, is limited from above by an input parameter as well as by the performance of the iterative solver. In addition to the **u**-solver (multigrid), also the Stokes solver (PC) is taken into account in the computation of δ. The numbers it_o of PC iterations in the time step and it_u of multigrid iterations to update **u** in the second Runge-Kutta step are evaluated, and the following criteria to change δ are applied to

these numbers:

$$\textbf{if } ((it \leq 1 \wedge \frac{||r_i||}{||r_0||} < 0.8\, tol) \vee (it \leq 2 \wedge \frac{||r_i||}{||r_0||} < 0.5\, tol)) \;\; \delta = \min(1.25\,\delta,\, \delta_{max}) \tag{5.4}$$

$$\textbf{if } (it_u \geq 5 \vee it_o \geq 7 \vee \frac{||r_i||}{||r_0||} > tol) \;\; \delta = 0.8\,\delta \tag{5.5}$$

$$\textbf{if } (\delta < \delta_{min}) \;\; \text{Stop} \tag{5.6}$$

Regarding the Schur complement equation, the norm of the unscaled divergence is used here. This keeps iteration numbers per time step roughly constant until δ_{max} is reached, leading to $\{it_u, it_o\} \in 1, 2$ in many cases. It also prevents the convection calculation from varying in accuracy as time steps are adapted rather quickly. Using the scaled divergence, however, can lead to alternating step sizes, this is observed when the accuracy of the inner solver for the velocity search direction is too low. Then the residual reduction per outer iteration is so low that it_o can increase rapidly. Using the unscaled divergence, time steps evolve more smoothly, although a bit slower than with the scaled divergence. Figure 5.3 shows the evolution of the advection step in the example cases, together with it_u and it_o. Note, that it_u cannot be less than one, because the multigrid solver always performs a v-cycle once it has computed the residual.

5.6 Convergence Results

All computations in this section have been done for 10 time steps to examine the effort to get a valid solution from a zero initial guess. So the initial conditions, shown in Figure 5.1 apply. As in Chapter 3, the main quantities of interest are the sum of inner iterations, it_i, here multigrid cycles throughout the first 10 steps, and the sum of pressure correction iterations, it_o. Note that in contrast to Chapter 3, the multigrid cycles required to obtain the **u**-solution are included in it_i. In Cases 200 and 901, it was also necessary to print out the number of multigrid calls where the specified residual reduction was not reached, as we must expect it_o to grow when this happens quite often. A comparison of all cases and of preconditioning with Jacobi iteration on the mass matrix against scaling is shown in Table 5.2. It shows the dependence of the iteration numbers on the stabilization weight γ.

The effect of scaling on the outer iteration count, it_o, can be seen very clearly in Figure 5.4. It is not surprising that there is essentially no difference in the constant viscosity Case 002, but already in Case 007 with its slight viscosity variation, residual reduction differs remarkably. The convergence of Case 200, however, suffers from the fact that the desired multigrid convergence tolerance is not achieved in many of the PC steps. Even though the convergence failure rate is higher with scaling (see

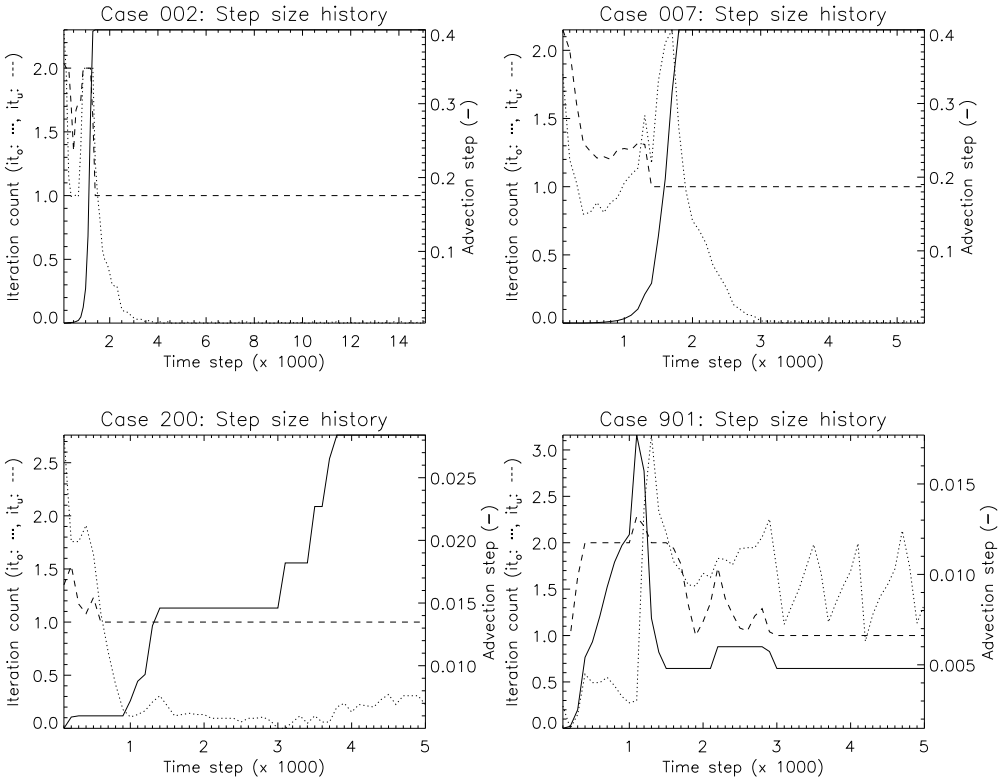

Figure 5.3: Evolution of advection step in the examples: Advection step δ (solid) and iteration counts it_u (dashed) and it_o (dotted) are averaged over intervals of 100 time steps. Calculations are done on grid level 6.

Table 5.2), the overall PC convergence is nevertheless more rapid *with* scaling. The frequent jumps in the residual to larger values for Case 200, evident in Figure 5.4, are mostly associated with PC restarts, where the **u**-recomputation results in an increase of the divergence of **u**. A tremendous benefit of scaling can be seen in Case 901 in which the lateral viscosity variation is strongest. Here the convergence is only slightly worse than in Case 007, despite the fact that the multigrid solver often fails to achieve the desired tolerance. Case 901 also reveals that scaling leads to subsystems for the velocity search directions which are more challenging to the multigrid solver that those of the unscaled system. The benefit of scaling for it_o, which decreases by a factor of 10-15, is notably larger than it is for it_i, which decreases it "only" by a factor of 4. However, if the performance of the multigrid solver could be improved, we would likely see a similar reduction in it_i also.

Table 5.2 also shows how iteration numbers depend on the stabilization weighting γ. In most cases, the dependency is minor, thus no plot is provided. As expected, a high γ leads to slightly higher iteration counts as the maximum divergence error from the discretization is already close to the stopping threshold. Only Case 200 on grid level 8 shows unexpected

Table 5.2: Summed iteration counts of example cases for the first 10 time steps on three grid levels. Stabilization weighting γ is varied from $\frac{1}{4}\gamma_{max}$ to γ_{max}. f denotes the number of u_0- and v-computations where multigrid failed to reach the specified stopping tolerance.

l	6				7				8			
Ex.	it_o	it_i	f	γ	it_o	it_i	f	γ	it_o	it_i	f	γ
Solve (5.1) with M as Schur complement preconditioner												
002	11	85	0	.005	8	69	0	.015	5	66	0	.05
002	12	86	0	.02	8	71	0	.06	5	67	0	.2
007	17	127	0	.003	19	137	0	.006	22	169	0	.02
007	32	186	0	.01	24	154	0	.025	28	203	0	.09
200	121	979	16	.002	133	1349	46	.013	178	918	46	.05
200	145	1272	33	.007	128	1157	34	.05	301	1952	53	.2
901	156	710	0	.008	188	875	0	.035	249	1100	0	.13
901	154	701	0	.015	201	930	0	.07	235	1054	0	.26
901	155	702	0	.03	210	974	0	.14	210	944	0	.5
Solve (5.2) with Schur complement scaled by $Y = 1/diag(M_\eta)$ with η_{HARM}												
002	14	98	0	.005	8	72	0	.015	5	65	0	.05
002	15	101	0	.02	8	72	0	.06	7	75	0	.2
007	21	140	0	.003	11	106	0	.006	8	100	0	.02
007	32	215	0	.01	14	124	0	.025	9	105	0	.09
200	91	1003	24	.002	64	1218	52	.013	42	766	29	.05
200	104	1157	29	.007	64	1227	51	.05	97	1625	61	.2
901	15	312	6	.008	13	333	11	.035	13	328	12	.13
901	15	312	6	.015	13	333	12	.07	13	329	12	.26
901	20	406	8	.03	19	458	18	.14	16	395	15	.5

high iteration numbers for $\gamma = \gamma_{max}$. This might be due to the poor multigrid performance in that case.

Table 5.3 and Figure 5.5 show the dependency of the iteration counts it_i and it_o on the specified accuracy of the inner solver and other iteration-related parameters. These are the maximal number of multigrid iterations *itmg* and the maximal allowed convergence number for continuing multigrid iterations *stopmg*. For a detailed description of these parameters, see Section B.1. Table 5.3 also contains rather "aggressive" parameter settings regarding accuracy, but as they are often used in practice, they are included in this study. They clearly show lower inner iteration counts it_i, but what is not seen here is that with these settings, the stopping thresholds for velocity and pressure are often reached as late as in the third to fifth time step. When considering that the Stokes system will become more difficult to solve during the convection calculation if temperature-dependence of viscosity is not strongly damped (see Figure 5.2, choosing the iteration parameters as low as possible will not turn out to be successful. Residual reduction is oscillating much more for the loose inner accuracy than for the one that is chosen as a standard in this work.

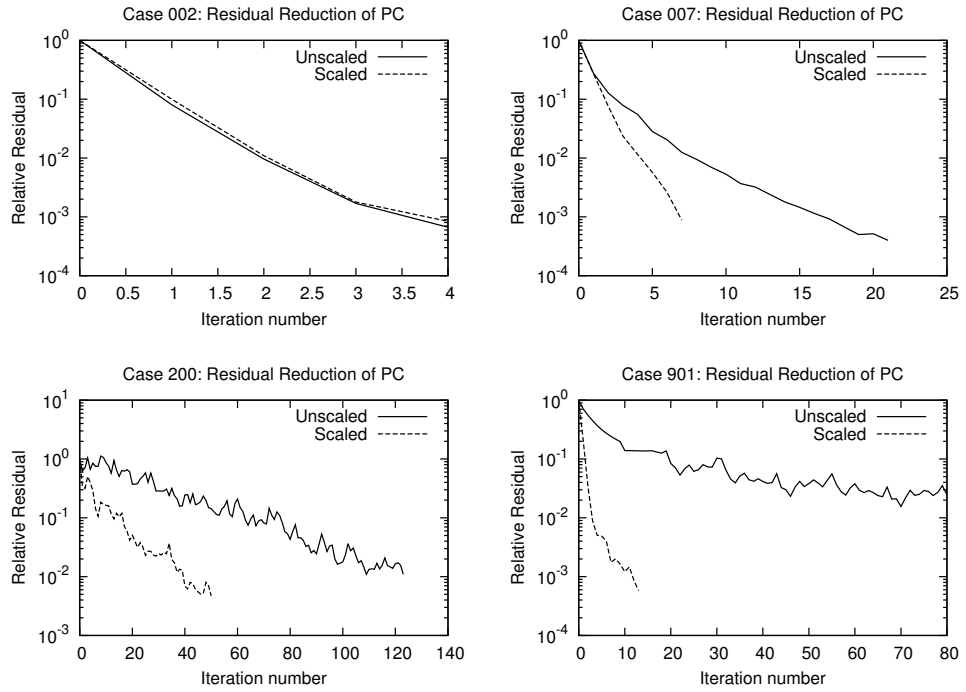

Figure 5.4: Comparison of it_o in the first Runge-Kutta step for unscaled (solid) and scaled (dashed) Schur complement: Computations are done on grid level 8 with $\gamma = \gamma_{max}$.

Table 5.3: Same as Table 5.2, here always with $\gamma = \frac{1}{4}\gamma_{max}$, but with different inner accuracy *vtol*: $vtolfac$ is denoted vtf, *itmg* is denoted it and *stopmg* is denoted stmg

l	6			7			8			vtf	it	stp
Ex.	it_o	it_i	f	it_o	it_i	f	it_o	it_i	f			
Solve (5.2) with S scaled by $Y = 1/diag(M_\eta)$ with η_{HARM}												
200	421	3693	103	98	1058	28	88	1079	38	5.0	20	0.99
200	240	898	41	93	433	61	110	483	52	5.0	20	0.95
200	120	998	121	91	524	89	117	689	109	2.0	20	0.95
901	15	240	3	13	296	6	13	312	8	5.0	20	0.99
901	15	198	1	13	248	3	13	263	6	10.0	20	0.99
901	16	165	1	13	176	3	13	193	2	30.0	20	0.99
901	15	168	3	14	148	3	13	149	3	100.0	20	0.99
901	15	122	5	14	128	6	13	128	6	100.0	10	0.99
901	15	122	5	14	122	9	13	126	7	100.0	10	0.95
Solve (5.2) with S scaled by $Y = 1/diag(M_\eta)$ with η_{GEOM}												
200	172	545	23	105	428	45	161	670	56	5.0	20	0.95
901	16	156	2	14	144	4	14	151	3	100.0	20	0.99
Solve (5.1) with M as Schur complement preconditioner												
200	169	536	49	145	427	16	161	468	23	5.0	20	0.95
901	152	419	0	225	606	2	217	580	1	100.0	20	0.95

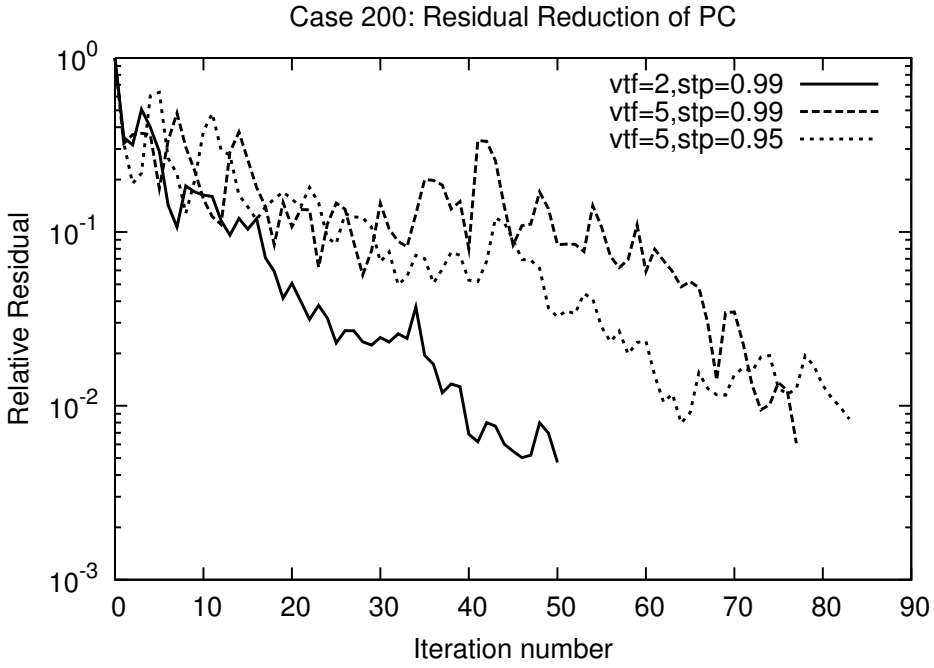

Figure 5.5: Schur complement residual reduction for Case 200 on grid level 8 for different inner accuracies (see also Table 5.3)

In Table 5.3 two rows are included, showing iteration numbers for the cases with strong viscosity variations when stabilization and mass matrices use 2-point geometric averaging of viscosity (see also Section 4.3.1). For Case 901 they stay essentially unchanged, as viscosity varies rather smoothly. Also for Case 200, they do not vary much. It seems that on grid level 6 the geometric mean is slightly advantageous, that they equal on grid level 7 and that on grid level 8 the harmonic averaging gives slightly better results. For grid level 8, the residual reduction for both methods is shown in Figure 5.6. They almost equal in the beginning, only after 50 iterations the residual is not further reduced with geometric averaging. A piece of information that is missing in both, Table 5.3 and Figure 5.6, is that using geometric mean requires the stabilization weight to be decreased by a factor of 14 for Case 200, compared to the harmonic mean, thus making it less attractive, the more as it does not give a clear benefit in the convergence of the PC algorithm.

5.7 Discussion and Bibliographical Notes

The stabilization weight γ in the four test cases varied between 0.02 and 0.5 for grid level 8. This is still lower than I what had expected from the results of Chapter 3. One possible explanation is related to the investigation of the accuracy of finite-element Stokes system solutions by Moresi

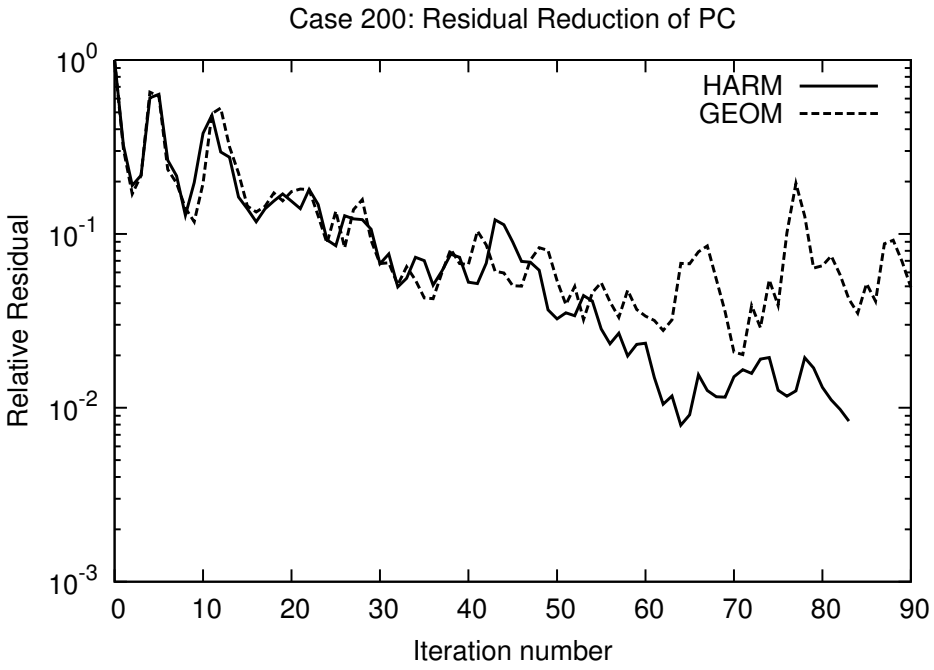

Figure 5.6: Dependency of Schur complement residual reduction for Case 200 on the viscosity averaging method on grid level 8.

et al. (1996). They found that, after the lowest possible pressure error has been reached, the discrete incompressibility could still be further reduced by several orders of magnitude without any effect to the pressure solution, because the discretization error level has already been reached. So it could also be the case that the discrete incompressibility is "overreduced" in the Terra-solver and the attainable pressure accuracy has already been reached. It was, however, noted by Dohrmann and Bochev (2004) that for some 3D elements $\gamma < 1$ gives better pressure accuracy and lower maximum divergence errors. To this note they gave neither further explanation nor detailed analysis as the results in their examples were satisfactory with $\gamma = 1$. This was also true in the Example 2 of this thesis (see Chapter 2) which was adopted from one of the examples in (Dohrmann and Bochev, 2004).

The anticipated benefit of applying pressure stabilization to the performance of the iterative solver (Dohrmann and Bochev, 2004; Schunk et al., 2002) could not be confirmed in the Terra code. Whereas in the examples of Chapter 2 both, the eigenvalue distribution of the Schur complement S and the iteration numbers of the pressure solver, strongly benefited from applying the stabilization, the effect on the eigenvalues could not be measured in Terra as no eigenvalue estimates were available. It is, however, not mandatory that a better eigenvalue distribution leads to an improved performance of the iterative solver. Although very often, unfavorable eigenvalues do not in any case slow down the solver's convergence

so that by eliminating them from the system an improved convergence is not guaranteed. There is no proposition and even no general assumption that stabilization in any case improves the solvers convergence.

The use of the variable-viscosity scaled pressure mass matrix M_η as a preconditioner for S leads to iteration numbers of the pressure correction algorithm only weakly dependent on the viscosity contrast. The larger the variation is, the more beneficial is the use of M_η instead of M. The use of a viscosity-dependent preconditioner is very common in developing variable-viscosity Stokes solvers. Moresi et al. (1996) use $B^T(\mathrm{diag}[\mathbf{A}])^{-1}B$ or only its diagonal. Such a preconditioner also contains the viscosity variations present in $\mathbf{A}$ and in S through the use of the diagonal of $\mathbf{A}$. The application of $\mathrm{diag}\left[B^T(\mathrm{diag}[\mathbf{A}])^{-1}B\right]$ is of the same cost as the application of $\mathrm{diag}[M_\eta]$ but its computation, which has to be done with every viscosity update, is a bit more expensive. The preconditioner $\mathrm{diag}\left[B^T(\mathrm{diag}[\mathbf{A}])^{-1}B\right]$ is also a bit less accurate as the reduction of a matrix to its diagonal is done twice compared to once in case of $\mathrm{diag}[M_\eta]$. Tackley (2008), who uses multigrid also for the pressure solution, employs a kind of matrix-dependent transfer for the pressure which adjusts pressure prolongations by a sort of weighted average of local viscosity values. Also here, the idea of a viscosity-dependent preconditioner is present.

The application of the adaptive solver restart criterion (5.3) follows the line of thinking from Verfürth (1984) and Peters et al. (2005). It gives strongly improved convergence of the PC algorithm in cases where multigrid often fails to reach the specified stopping tolerance in solving the velocity subsystem. In cases where where multigrid works efficiently it is less important but still beneficial to use such an intermediate stopping criterion. The numerical values in (5.3) have been optimized to yield the best convergence with the current Terra code, but the principle does not depend on the specific convergence properties of the inner solver.

It could be confirmed with the test cases used in this chapter that the capabilities of the current multigrid implementation in Terra set the limit for the viscosity model to be used in the convection calculation. Using strong gradients of coefficients is not easy with multigrid (Trottenberg et al., 2001). In mantle convection simulations, Choblet (2005) reports convergence problems with strong viscosity gradients rather than with the global contrast for a spherical FV-discretization with a multigrid solver. Kameyama et al. (2005) used an increased number of pre- and post- smoothing steps to get a satisfactory multigrid convergence with strongly varying viscosity. Without mentioning explicitly the limits of their multigrid implementations, most researchers limit their global viscosity variation to 5 to 7 orders of magnitude (Yoshida, 2004; Yoshida and Nakakuki, 2009; Stemmer et al., 2006; Zhong et al., 2007), which is sufficient if compositional heterogeneities in the mantle are neglected. As these are present, also strong gradients in viscosity need to be considered. I am aware of only one other spherical-shell model using viscosity gradi-

ents comparable to those of Walzer et al. (2004b) which are used in Case 200. This model (Tackley, 2008) shows that a multigrid algorithm can indeed be used to solve such a problem by achieving almost constant iteration numbers up to a global viscosity contrast of 10 orders of magnitude. With 15 pre- and post-smoothing steps a convergent solution for a global contrast up to 19 orders of magnitude and local contrasts of up to 54,000 for adjacent grid points could be obtained. This, together with the successful application of diagonal scaling of $\mathbf{A}$ in Chapter 2, indicates that a significant improvement of the multigrid implementation in Terra is feasible. With diagonal scaling of $\mathbf{A}$, most of the viscosity variation should be scaled out of $\mathbf{XAX}$, and the use of cell-wise harmonic viscosity averaging smooths the numerical representation of nodal viscosity contrasts a bit. So a significant improvement of the velocity subsystem solves in Terra's PC algorithm can be expected.

Chapter 6

Summary and Conclusions

In the previous chapters, polynomial pressure projections have been used to stabilize a $Q_1 - Q_1$ finite-element discretization of the Stokes equations in a two-dimensional square domain as well as in a three-dimensional spherical shell. Beside suppressing spurious pressure oscillations, this stabilization strongly improves the spectral properties of the Schur complement $S = B^T \mathbf{A}^{-1} B + C$ compared to those of $B^T \mathbf{A}^{-1} B$. This could be confirmed numerically for the square domain in Chapter 2 for both, constant and variable viscosity. While viscosity in the square-domain examples was assumed to be element-wise constant, in the Terra code cell-wise harmonic averages of the nodal viscosity field were used in constructing the stabilization matrix C.

A diagonal scaling has been applied to the system matrices $\mathbf{A}$ and S using the diagonal entries of $\mathbf{A}$ and of the viscosity-dependent pressure mass matrix M_η, respectively. While this could also be considered as part of a preconditioning strategy, it filters out most of the viscosity variation from the whole Stokes system. Convergence of the outer (pressure) solver could be significantly improved with this scaling.

A comparison of three suitable Krylov subspace methods has been carried out in the square domain for different viscosity structures, viscosity contrasts and for different accuracies of the inner solver. The examined solvers are the pressure correction algorithm (PC), minimum residual algorithm (MINRES) and a conjugate gradient algorithm modified by Bramble and Pasciak (1988) (BPCG). Their convergence behavior has been investigated using two analytical solutions with Dirichlet boundary conditions.

Stopping criteria for the iterative solvers have been systematically varied in the two-dimensional analytical examples to yield optimal values to minimize the computational cost for achieving an iterative solution close to the accuracy of the underlying discretization. Furthermore, the robustness of the above-mentioned three Krylov solvers regarding variations of the stopping criteria has been examined. In the Terra code, stopping criteria have been varied and an adaptive criterion, given in (5.3), has been developed.

Within the investigation of the pressure solver, the inner (velocity) solver in Terra could be identified to be not efficient enough for sustaining iteration numbers almost independent of the viscosity contrast. Especially strong gradients of viscosity lead to unsatisfactory poor convergence of the multigrid solver as it is currently implemented into Terra. It should be mentioned that the rather poor viscosity averaging using the geometric mean of only two points of a grid cell has not been changed to a cell-wise harmonic mean of all nodal values belonging this cell during this work. This is subject to a further improvement of Terra's capabilities to include viscosity variations.

The treatment of the free-slip boundary condition in the Terra code has been left unchanged during this thesis work. As the formulation of this condition on a curved surface is very cumbersome using Cartesian coordinates, the team of the Terra-developers currently works on an efficient and precise formulation. Although the constant viscosity convergence is satisfactory also for the multigrid solver, it might be the case that it can still be improved by a more efficient formulation of the free-slip condition.

From the research described above, the conclusions can be drawn as follows:

Pressure stabilization The stabilization technique of Dohrmann and Bochev (2004), using local polynomial pressure projections, could be implemented into Terra. Its effect in grabbing pressure solutions with non-vanishing continuous but vanishing discrete gradient becomes stronger with ever finer grid resolution. However, on coarse grids the increase of the discretization error due to the piecewise constant pressure requires the stabilization matrix to be decreased in its influence. On grid level 8 and finer, it can be applied almost without a decreasing factor.

In SSST, the effect of the stabilization on the iterative solver was rather strong. This is not the case in Terra. However, there is also no negative influence if the gap between discretization error and the allowance for the iteration error is not too small.

Scaling Using the viscosity-dependent mass matrix M_η to scale the Schur complement with $\mathrm{diag}(M_\eta)$ strongly improves the performance of any Krylov solver in cases with strongly varying viscosity.

However, it requires the observation also of unscaled norms to make sure that the right quantity is reduced by as much as we want. But the overhead by caring for a second norm is far outweighed by the much better convergence when using scaled matrices.

Krylov solver From the study of the three solvers PC, MINRES and BPCG it became clear that the difference between between efficient implementations of suitable Krylov solvers is rather small. In most cases it does not justify switching from an existing implementation, tailored to a specific problem, to another Krylov method. If a "best"

solver should be chosen, considering efficiency, robustness and implementational effort, PC is recommended, especially for high viscosity variations. This is also in accordance with the results of ur Rehman (2009).

Solver restart With appropriate restart criteria, also the pressure correction method can converge almost uniformly, even though a low accuracy in the solution of the velocity search directions is enforced. The effort in choosing appropriate restart criteria is moderate as the sensitivity to these is not too high.

Multigrid Although not considered in detail here, the results in Chapter 5 suggest that the multigrid method is the "bottleneck" in Terra's solver. Thus, together with providing harmonic viscosity averaging also in the **A**-operator, investigating the multigrid solver should be one of the next tasks in further improving the Terra code.

Appendix A

Further Information to SSST

A.1 Detailed Results for Example 2

The results for Example 2 are quite similar to those for Example 1. However, two minor differences can be observed: MINRES is less efficient for exponential viscosity structures and BPCG.R is less efficient for high viscosity inclusions compared to Example 1. To find the explanation for this, let us consider the differences in problem description of the two Examples. In addition to a smaller pressure gradient, which can be seen in Figure 2.2, Example 2 has a non-vanishing right-hand side and a smaller domain, halved in both dimensions. These differences can be the cause for a non-optimal weighting of $\mathbf{u}$- and p-errors as well as for a non-optimal scaling of the S and a too large k_{bp} in BPCG.R. Moreover, it was necessary to set the upper limit of *bptol* from 0.2 down to 0.03 to prevent it_i for I12 from exceeding 200000. Only PC.R performs independent of the scaling of S, which is also shown in Peters et al. (2005, Table 5).

Table A.1: Iteration numbers of PC for Example 2

l	4		5		6		7		8	
Visc.	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i
E04	19	213	19	478	22	1168	23	2343	25	5138
E08	39	235	39	478	44	1095	52	2787	53	5920
E12	48	215	73	575	78	1272	66	2222	76	5534
S04	19	316	20	694	21	1406	24	3258	27	6472
S08	27	490	31	1073	33	2120	35	4498	37	9200
S12	32	498	37	1097	39	2409	43	5672	37	8837
I04	22	819	20	1749	19	3911	18	8080	19	17575
I08	31	2001	26	3644	27	8456	26	17444	26	35141
I12	40	3076	36	5890	33	11898	34	27761	33	57525

Table A.2: Iteration numbers of PMINRES for Example 2

l	4		5		6		7		8	
Visc.	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i
E04	43	441	45	974	46	2138	48	4772	48	10152
E08	60	606	71	1411	79	3388	74	6518	76	14241
E12	76	776	96	1912	103	4290	103	8886	100	17847
S04	51	491	74	1138	55	2037	48	3857	52	9328
S08	73	708	85	1455	75	2806	68	5540	69	11608
S12	72	677	78	1293	69	2556	68	5834	70	13272
I04	73	2472	45	3303	40	6689	43	13875	49	37897
I08	150	7614	69	8078	60	16426	52	27369	53	62432
I12	67	8134	61	14341	63	28901	64	67027	61	140261

Table A.3: BPCG.R iterations: $k_{bp} = \min(\max(1 - 2kh\sqrt{\kappa(S_5)}, 0.8), 0.99)$

l	4		5		6		7		8	
Visc.	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i	it_o	it_i
E04	45	371	46	718	37	1395	29	2614	33	6722
E08	42	243	57	595	62	1220	70	2984	65	6172
E12	71	299	99	773	109	1646	117	3170	119	7325
S04	27	362	35	838	28	1646	31	4128	27	8423
S08	47	595	43	1025	33	1793	39	5082	34	9570
S12	47	563	61	1329	42	2473	47	5996	40	11369
I04	25	1193	24	2540	17	4194	26	15099	24	26390
I08	25	2568	26	5507	23	10467	27	25538	27	51538
I12	62	7542	43	11718	37	23657	36	45788	47	124170

Table A.4: Stopping criteria for all solvers in Example 2

Solver	PC.R		MINRES		BPCG.R	
Visc.	tol	$itol$	tol	$itol$	tol	$itol$
E04	10^{-5}	10	10^{-6}	0.1	10^{-4}	1
E08	10^{-6}	1	10^{-6}	0.01	10^{-5}	1
E12	10^{-6}	1	10^{-7}	0.001	10^{-6}	1
S04	10^{-5}	10	10^{-6}	1	10^{-4}	1
S08	10^{-7}	10	10^{-8}	1	10^{-5}	1
S12	10^{-8}	10	10^{-8}	1	10^{-6}	1
I04	10^{-3}	0.1	10^{-4}	1	1	1
I08	10^{-2}	0.1	10^{-6}	1	10	0.1
I12	10^{-1}	0.1	10^{-8}	0.1	10	1

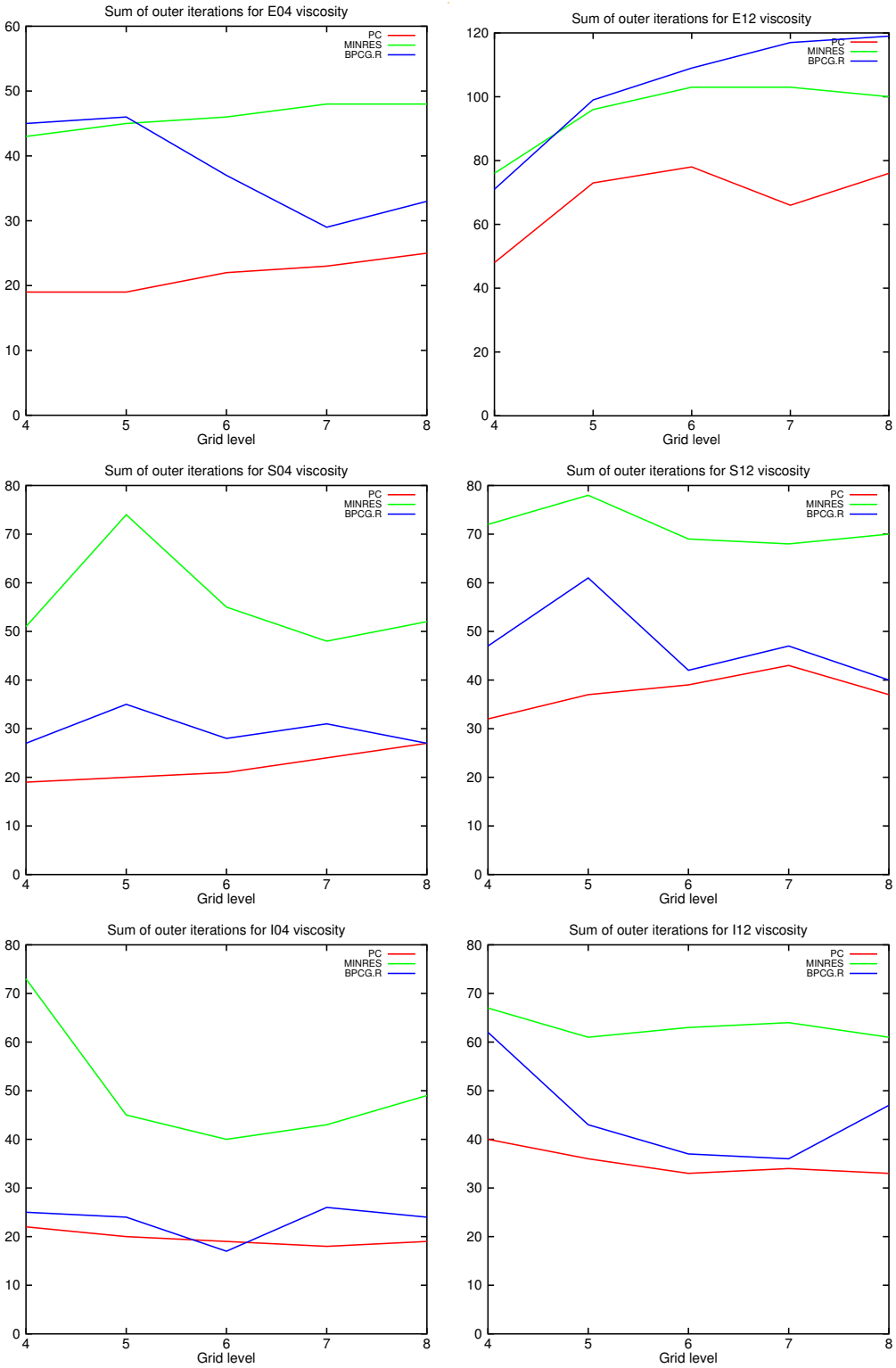

Figure A.1: Sum of outer iterations for all solvers in Example 2

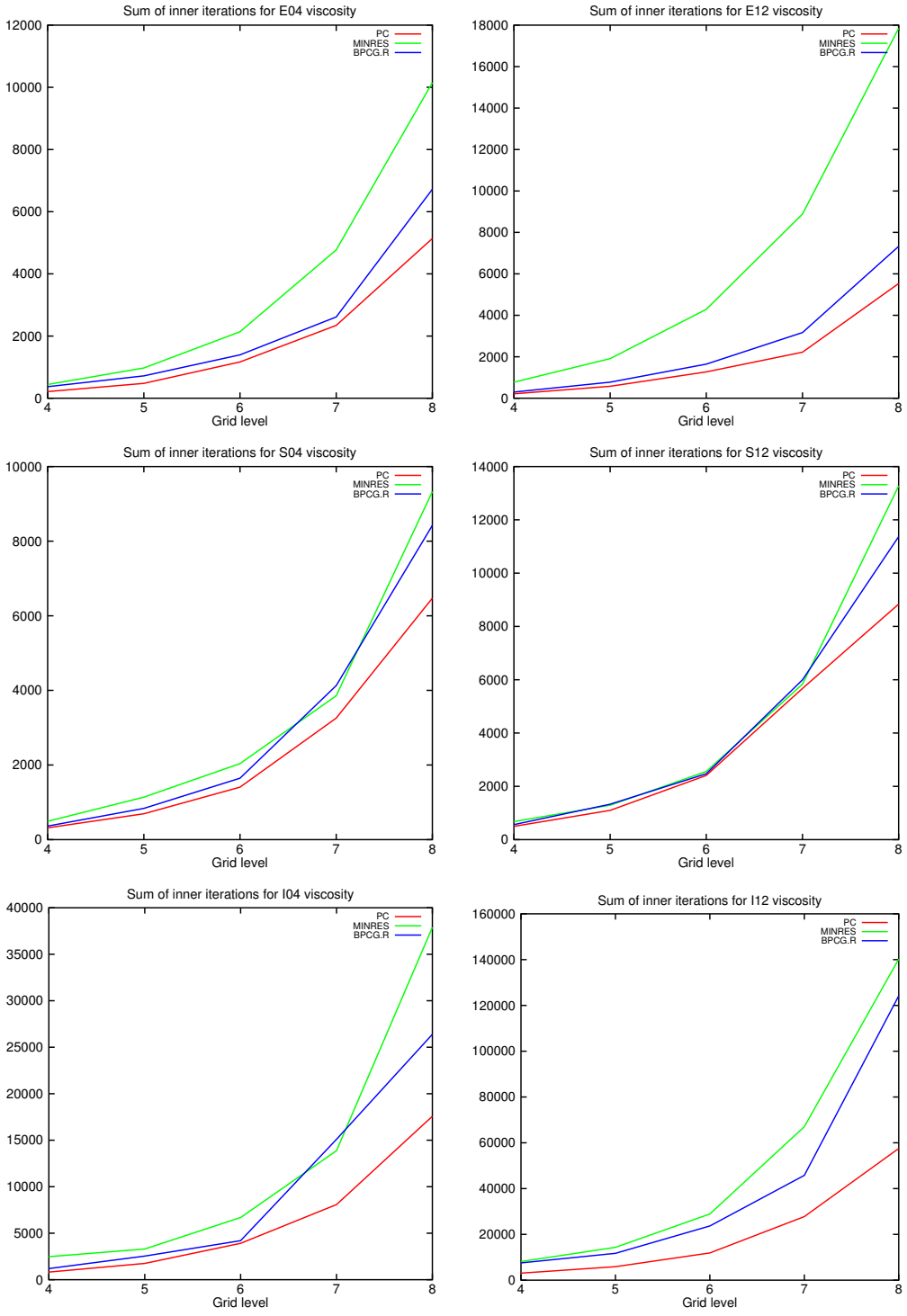

Figure A.2: Sum of inner iterations for all solvers in Example 2

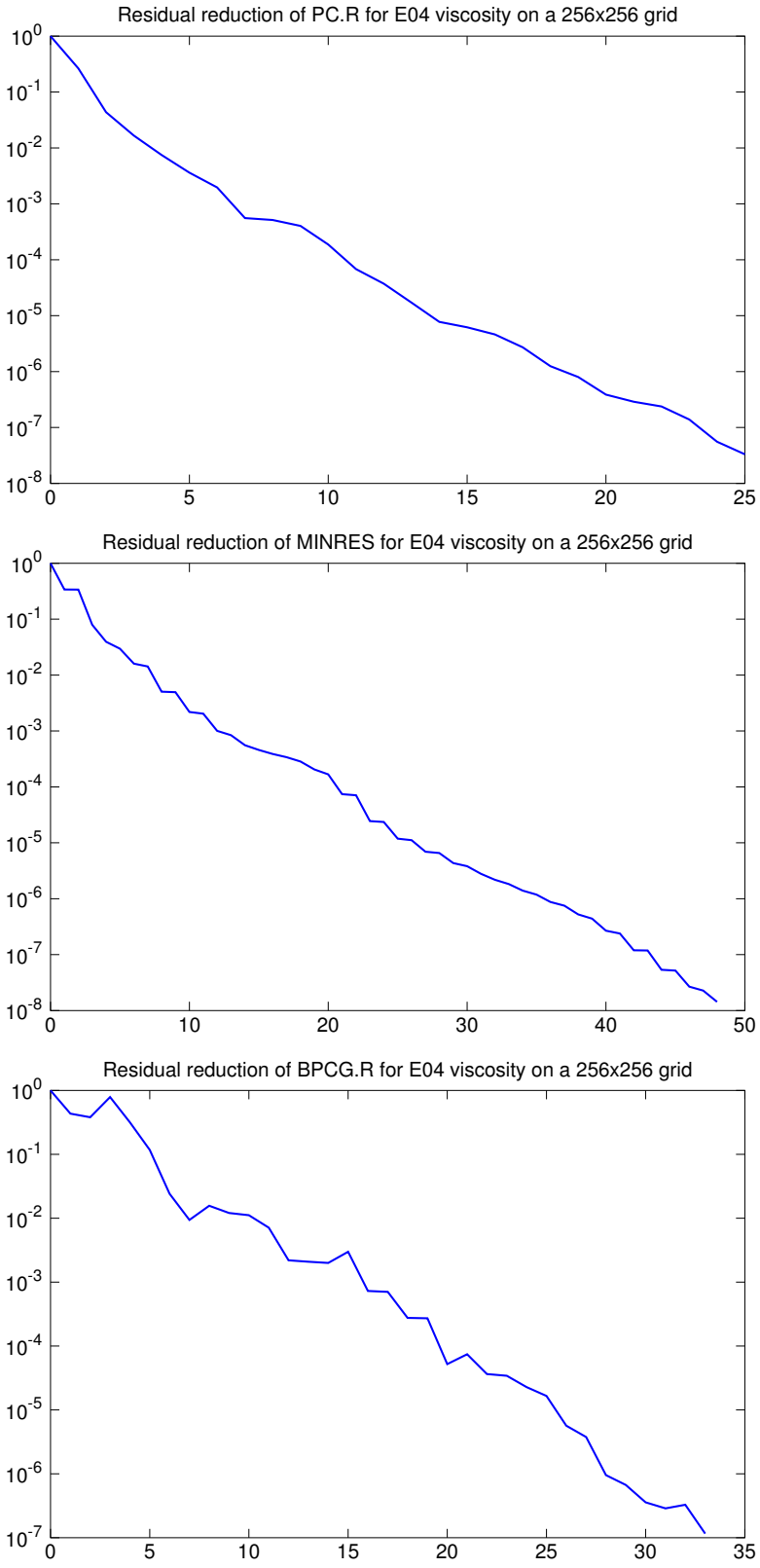

Figure A.3: Residual reduction for E04 in Example 2

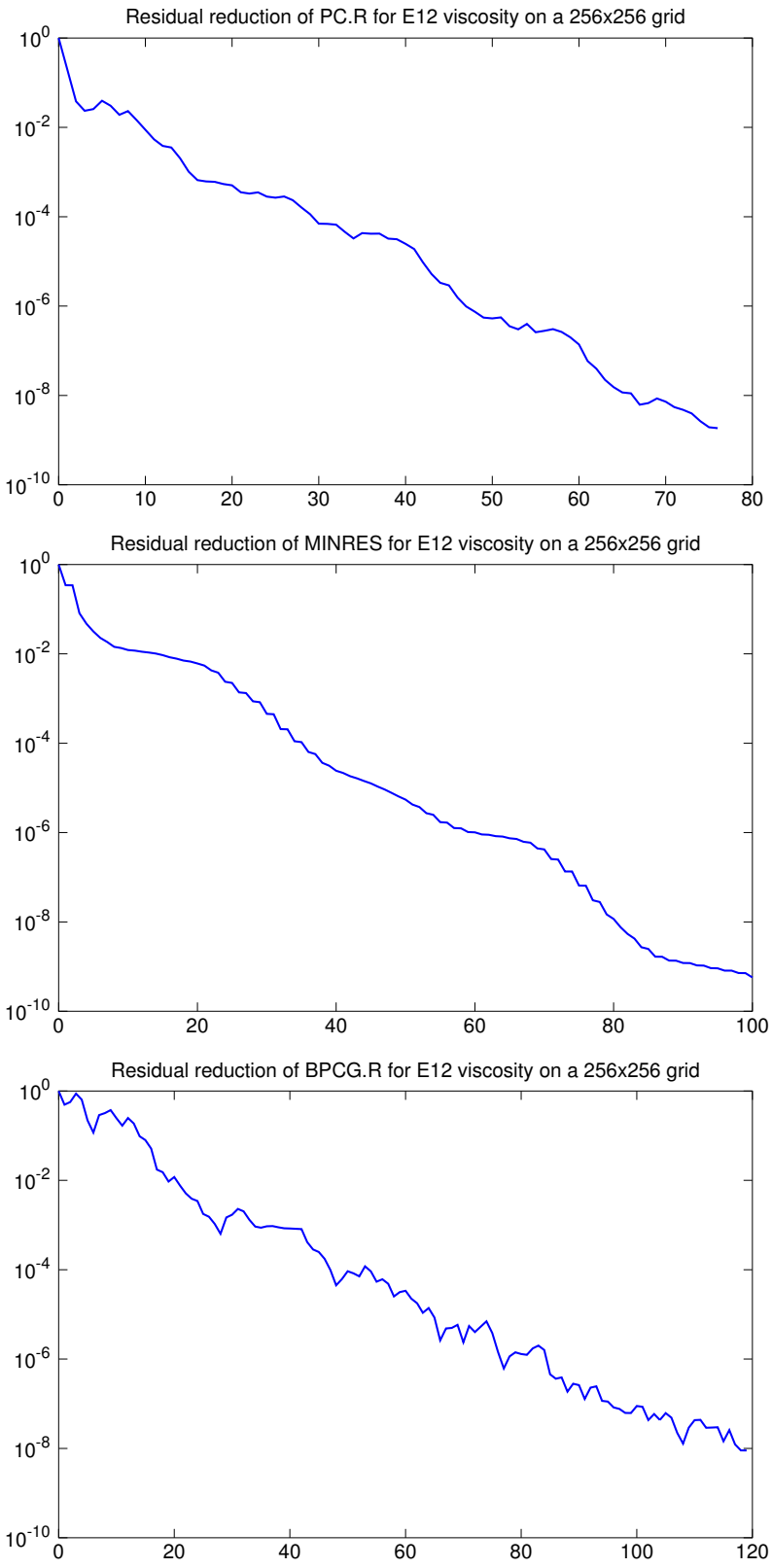

Figure A.4: Residual reduction for E12 in Example 2

Appendix B

Further Information to Terra

B.1 Input Parameters in the Code

As some parameters to control the solver were fixed in the code before, I added them to the list of input parameters. Data types are set implicitly by Fortran.

stabmax This parameter serves as an initial value and as maximum value of the stabilization weight γ, which is computed adaptively. Currently the code chooses γ as high as possible, but the findings of Section 5.6 suggest changing the code to choose γ to be half the maximum value. If *stabmax* is set too high, it only leads to some extra PC iterations in the first time step(s) until the optimal value is found. If set too low, no adaption in the first time step is necessary, but γ cannot adapt as the Stokes system changes in later time steps. So, at least for grid level 8 and higher, it should be set to 1.0. The current value of γ is printed with other solver information to the out-file.

mprec This is a switch to control how the Schur complement should be preconditioned or scaled. It can take the following values:

- **1** Use constant-viscosity mass matrix and apply its inverse with 5 Jacobi iterations.
- **2** Use diagonal of constant-viscosity mass matrix and apply its inverse.
- **3** Use diagonal of variable-viscosity mass matrix and apply its inverse.
- **4** Use scaling with the square root of the diagonal of the variable-viscosity mass matrix.

Value 4 is the default and is highly recommended for variable-viscosity cases, based on the results in Section 5.6.

utol This sets the required residual reduction for $\mathbf{u}_0$ computations. Whereas **u** from the previous Runge-Kutta step is taken as initial guess, *utol* is multiplied with the residual from a zero initial guess to obtain the stopping residual. However, in any case at least one multigrid v-cycle is executed as the residual compuation is done at the beginning of the cycle. This parameter is a new name for the parameter formerly named *convtol*.

ptolfac The factor multiplying *utol* to specify *ptol*, the required residual reduction for solving the Schur complement equation with a pressure correction (PC) algorithm.

vtolfac The factor mulitplying *utol* to specify *vtol*, the required residual reduction for computing the velocity search directions in PC, which is also done using multigrid.

itstokes Maximum number of PC iterations in a single Runge-Kutta step. It is important mainly in the beginning of the convection calculation where the solution is sought with a zero initial guess. Note that PC itself is run with an inner-outer scheme with an adaptive restart criterion. To make use of it, $itstokes \geq 30$ should be chosen. In most cases, this is sufficient if scaling of the Schur complement is used (see Section 5.6). The upper limit, however, based on the current settings of N_i and N_o in Algorithm 6 is $itstokes = 200$.

itmg Maximum number of multigrid iterations. With the current implementation, it was found that it should not be set to a value larger than 20 or 30. A value between 10 and 20 seems to be optimal. This parameter is a new name for the parameter formerly named *itlimit*.

stopmg This parameter can be used to stop multigrid from doing further V-cycles if the convergence number of the last cycle has already been above stopmg. Default is 0.99.

B.2 Parallelization

While mentioned only briefly in this dissertation, discretization and solution in Terra are parallelized using explicit message passing with MPI. Therefore, it was important to choose a stabilization technique which can be applied locally. This is the case for the stabilization matrix using local pressure projections. The switch from mass-matrix preconditioning to diagonal scaling was beneficial not only in terms of iteration numbers but also in terms of communication as a diagonal matrix can be computed without communication. Only the viscosity averaging requires message passing. To illustrate the parallelization, Figure B.1 is given.

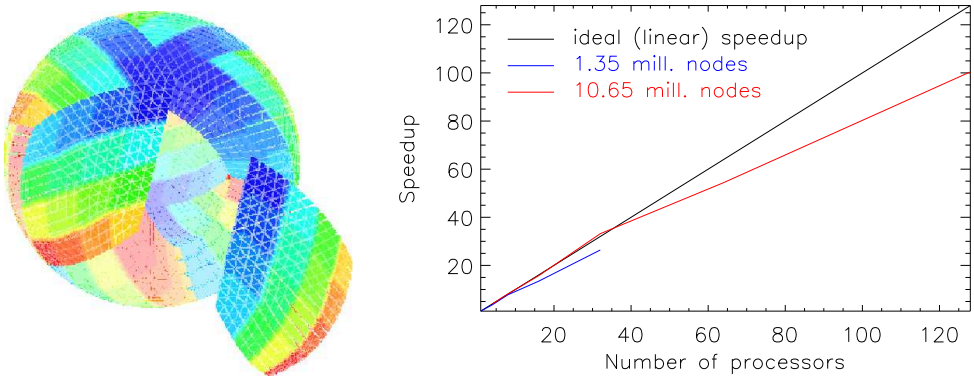

Figure B.1: Left: Illustration of domain decomposition in Terra. Every color represents a MPI process. Source unknown. Right: Scalability of Terra: Computed for grid levels 6 and 7 on the HLRB2 (SGI Altix 4700) at National Supercomputing Center LRZ Garching.

Bibliography

J. R. Baumgardner. *A Three-Dimensional Finite Element Model for Mantle Convection*. PhD thesis, Univ. of California, Los Angeles, 1983.

J. R. Baumgardner. Three-dimensional treatment of convective flow in the Earth's mantle. *Journal of Statistical Physics*, 39:501–511, 1985.

J. R. Baumgardner and P. O. Frederickson. Icosahedral discretization of the two-sphere. *SIAM J. Numer. Anal.*, 22:1107–1115, 1985.

D. Bercovici. The generation of plate tectonics from mantle convection. *Earth Planet. Sci. Lett.*, 205:107–121, 2003.

D. Bercovici. Mantle dynamics past, present and future: An introduction and overview. In D. Bercovici, editor, *Treatise on Geophysics, Vol. 7: Mantle Dynamics*, pages 1–30. Elsevier, 2007.

M. I. Billen. Modeling the dynamics of subducting slabs. *Annu. Rev. Earth Planet Sci.*, 36:325–356, 2008.

N. Boal, V. Domínguez, and F. Sayas. Asymptotic properties of some triangulations of the sphere. *Journal of Computational and Applied Mathematics*, 211:11–22, 2008.

P. B. Bochev and R. B. Lehoucq. Regularization and stabilization of discrete saddle-point variational problems. *Electronic Transactions on Numerical Analysis*, 22:97–113, 2006.

P. B. Bochev, C. R. Dohrmann, and M. D. Gunzburger. Stabilization of low-order mixed finite elements for the Stokes equations. *SIAM Journal on Numerical Analysis*, 44:82–101, 2006.

C. I. Bovolo. The physical and chemical composition of the lower mantle. *Phil. Trans. Royal Soc. A: Math. Phys. Engng. Sci.*, 363:2811, 2005.

J. H. Bramble and J. E. Pasciak. A preconditioning technique for indefinite systems resulting from mixed approximations of elliptic problems. *Math. of Computation*, 50(181):1–17, 1988. ISSN 00255718.

F. Brezzi. On the existence, uniqueness and approximation of saddle-point problems arising from Lagrangian multipliers. *RAIRO Anal. Numer*, 8 (2):129–151, 1974.

H.-P. Bunge and J. R. Baumgardner. Mantle convection modelling on parallel virtual machines. *Computers in Physics*, 9:207–215, 1995.

H.-P. Bunge, M. A. Richards, and J. R. Baumgardner. A sensitivity study of three-dimensional spherical mantle convection at 10^8 Rayleigh number: Effects of depth-dependent viscosity, heating mode and an endothermic phase change. *J. Geophys. Res.*, 102:11991–12007, 1997.

C. Burstedde, O. Ghattas, G. Stadler, T. Tu, and L. C. Wilcox. Parallel scalable adjoint-based adaptive solution of variable-viscosity stokes flow problems. *Computer Methods in Applied Mechanics and Engineering*, 198(21-26):1691 – 1700, 2009. ISSN 0045-7825. doi: 10.1016/j.cma.2008.12.015. Advances in Simulation-Based Engineering Sciences - Honoring J. Tinsley Oden.

G. Choblet. Modelling thermal convection with large viscosity gradients in one block of the cubed sphere. *J. Comp. Phys.*, 205:269–291, 2005.

U. Christensen. Convection with pressure-and temperature-dependent non-Newtonian rheology. *Geophys. J. Royal Astr. Soc.*, 77:343–384, 1984a.

U. Christensen and H. Harder. Three-dimensional convection with variable viscosity. *Geophys. J. Int.*, 104:213–226, 1991.

U. R. Christensen. Heat transport by variable viscosity convection and implications for the Earth's thermal evolution. *Phys. Earth Planet. Int.*, 35:264–282, 1984b.

Y. Deubelbeiss and B. Kaus. Comparison of Eulerian and Lagrangian numerical techniques for the Stokes equations in the presence of strongly varying viscosity. *Phys. Earth Planet. Int.*, 171:92–111, 2008. ISSN 0031-9201.

C. Dohrmann and P. Bochev. A stabilized finite element method for the Stokes problem based on polynomial pressure projections. *Int. J. Num. Meth. Fluids*, 46:183–201, 2004.

H. C. Elman. Multigrid and Krylov subspace methods for the discrete Stokes equations. *Int. J. Numer. Methods Fluids*, 22:755–770, 1996.

H. C. Elman, D. J. Silvester, and A. J. Wathen. *Finite Elements and Fast Iterative Solvers: with Applications in Incompressible Fluid Dynamics*. Oxford Univ. Press, 2005.

T. Geenen, M. ur Rehman, S. P. MacLachlan, G. Segal, C. Vuik, A. P. van den Berg, and W. Spakman. Scalabale robust solvers for unstructured FE modeling applications; solving the Stokes equation for models with large, localized, viscosity contrasts. *Geochem. Geophys. Geosys.*, 10:1–12, 2009.

T. V. Gerya. *Numerical Geodynamic Modelling*. Cambridge Univ. Press, Cambridge, UK, 2010.

T. V. Gerya, J. A. D. Connolly, D. A. Yuen, W. Gorczyk, and A. M. Capel. Seismic implications of mantle wedge plumes. *Phys. Earth Planet. Int.*, 156:59–74, 2006. doi: 10.1016/j.pepi.2006.02.005.

T. V. Gerya, J. A. D. Connolly, and D. A. Yuen. Why is terrestrial subduction one-sided? *Geology*, 36:43–46, 2008. doi: 10.1130/G24060A.1.

G. H. Golub and C. F. Van Loan. *Matrix computations*. Johns Hopkins Univ. Press, 1996.

A. L. Hales. Convection Currents in the Earth. *Geophys. J. Int.*, 3:372–379, 1936.

H. Harder and U. Hansen. A finite-volume solution method for thermal convection and dynamo problems in spherical shells. *Geophys. J. Int.*, 161:522–532, 2005.

H. H. Hess. History of ocean basins. In *Petrologic studies*, pages 599–620. GSA, Boulder, CO, 1962.

A. W. Hofmann. Sampling mantle heterogeneity through oceanic basalts: Isotopes and trace elements. In R. W. Carlson, editor, *Treatise on Geochemistry, Vol.2: The Mantle and the Core*, pages 61–101. Elsevier, Amsterdam, 2003.

A. Holmes. Radioaktivität und die thermische Geschichte der Erde. *Naturwissenschaften*, 19:73–79, 1931.

A. Kageyama and T. Sato. "Yin-Yang grid": An overset grid in spherical geometry. *Geochem. Geophys. Geosys.*, 5:Q09005, 2004.

M. Kameyama, A. Kageyama, and T. Sato. Multigrid iterative algorithm using pseudo-compressibility for three-dimensional mantle convection with strongly variable viscosity. *J. Comp. Phys.*, 206:162–181, 2005.

M. Knapmeyer. Location of seismic events using inaccurate data from very sparse networks. *Geophys. J. Int.*, 175:975–991, 2008.

M. Larin and A. Reusken. A comparative study of efficient iterative solvers for generalized Stokes equations. *Numer. Linear Algebra Appl.*, 15(1): 13–34, 2008. doi: 10.1002/nla.561.

A. Loddoch, C. Stein, and U. Hansen. Temporal variations in the convective style of planetary mantles. *Earth Planet. Sci. Lett.*, 251:79–89, 2006.

D. A. May. Preconditioning variable viscosity Stokes flow problems associated with a stabilised finite element discretisation. In *11th International Workshop on Modeling of Mantle Convection and Lithospheric Dynamics*, Braunwald, 2009.

D. A. May and L. Moresi. Preconditioned iterative methods for stokes flow problems arising in computational geodynamics. *Phys. Earth Planet. Int.*, 171:33–47, 2008.

D. P. Mckenzie, J. M. Roberts, and N. O. Weiss. Convection in the Earth's mantle: Towards a numerical simulation. *J. Fluid Mech.*, 62:465–538, 1974.

A. Meyer and T. Steidten. Improvements and experiments on the Bramble-Pasciak type CG for mixed problems in elasticity. Technical Report Preprint SFB393/01-13, TU Chemnitz, 2001.

A. Meyer and P. Steinhorst. Überlegungen zur Parameterwahl im Bramble-Pasciak-CG für gemischte FEM. Technical Report Preprint SFB393/05-07, TU Chemnitz, 2005.

L. Moresi, S. Zhong, and M. Gurnis. The accuracy of finite element solutions of stokes's flow with strongly varying viscosity. *Phys. Earth Planet. Int.*, 97:83–94, 1996.

L.-N. Moresi and V. S. Solomatov. Numerical investigation of 2D convection with extremely large viscosity variations. *Phys.Fluids*, 7:2154–2162, 1995.

M. Müller. *Towards a robust Terra code*. PhD thesis, Friedrich-Schiller-Univ. Jena, http://www.igw.uni-jena.de/geodyn, 2008.

M. Olshanskii and A. Reusken. A Stokes interface problem: Stability, finite element analysis and a robust solver. In P. Neittaanmaki, T. Rossi, K. Majava, and O. Pironneau, editors, *European Congress on Computational Methods in Applied Sciences and Engineering ECCOMAS 2004*, 2004.

M. Olshanskii and A. Reusken. Analysis of a Stokes interface problem. *Numer. Math.*, 103:129–149, 2006.

C. C. Paige and M. A. Saunders. Solution of sparse indefinite systems of linear equations. *SIAM Journal on Numerical Analysis*, pages 617–629, 1975.

C. L. Pekeris. Thermal Convection in the Interior of the Earth. *Geophys. J. Int.*, 3:343–367, 1935.

J. Peters, V. Reichelt, and A. Reusken. Fast iterative solvers for discrete stokes equations. *SIAM Journal on Scientific Computing*, 27:646–666, 2005. doi: 10.1137/040606028.

D. Randall, T. Ringler, R. Heikes, P. Jones, and J. Baumgardner. Climate modeling with spherical geodesic grids. *Computing in Science & Engineering*, 4(5):32–41, 2002. ISSN 1521-9615. doi: 10.1109/MCISE.2002.1032427.

C. C. Reese, V. S. Solomatov, and J. R. Baumgardner. Scaling laws for time-dependent stagnant lid convection in a spherical shell. *Phys. Earth Planet. Int.*, 149:361–370, 2005.

G. C. Richard and D. Bercovici. Water-induced convection in the Earth's mantle transition zone. *J. Geophys. Res.*, 114:14–17, 2009.

M. A. Richards, W.-S. Yang, J. R. Baumgardner, and H.-P. Bunge. Role of a low-viscosity zone in stabilizing plate tectonics: Implications for comparative terrestrial planetology. *Geochem. Geophys. Geosys.*, 3:1040, 2001. doi: 10.1029/2000GC000115.

S. K. Runcorn. Paleomagnetic comparisons between Europe and North America. *Proc. Geol. Assoc. Canada*, 8:77–85, 1956.

G. Schubert, D. L. Turcotte, and T. R. Olson. *Mantle Convection in the Earth and Planets*. Cambridge Univ. Press, Cambridge, UK, 2001.

P. Schunk, M. Heroux, R. Rao, T. Baer, S. Subia, and A. Sun. Iterative solvers and preconditioners for fully-coupled finite element formulations of incompressible fluid mechanics and related transport problems. Technical report, Sandia National Labs., Albuquerque, NM (US), 2002.

S. V. Sobolev and A. Y. Babeyko. What drives orogeny in the Andes? *Geology*, 33(8):617–620, 2005.

V. S. Solomatov and C. C. Reese. Grain size variations in the Earth's mantle and the evolution of primordial chemical heterogeneities. *J. Geophys. Res.*, 113:B07408, 2008. doi: 10.1029/2007JB005319.

K. Stemmer, H. Harder, and U. Hansen. A new method to simulate convection with strongly temperature- and pressure-dependent viscosity in a spherical shell: Applications to the Earth's mantle. *Phys. Earth Planet. Int.*, 157:223–249, 2006.

M. Tabata. Finite element approximation to infinite Prandtl number Boussinesq equations with temperature-dependent coefficients–Thermal convection problems in a spherical shell. *Future Generation Computer Systems*, 22:521–531, 2006.

M. Tabata and A. Suzuki. A stabilized finite element method for the Rayleigh-Bénard equations with infinite Prandtl number in a spherical shell. *Computer Methods in Applied Mechanics and Engineering*, 190: 387 – 402, 2000. ISSN 0045-7825. doi: 10.1016/S0045-7825(00)00209-7.

M. Tabata and A. Suzuki. Mathematical modeling and numerical simulation of Earth's mantle convection. In I. Babuska, P. G. Ciarlet, and T. Miyoshi, editors, *Lecture Notes in Computational Science and Engineering*, volume 19, pages 219–232. Springer, 2002.

P. Tackley. Effects of strongly variable viscosity on three-dimensional compressible convection in planetary mantles. *J. Geophys. Res.*, 101 (B2):3311–3332, 1996.

P. J. Tackley. Effects of strongly temperature-dependent viscosity on time-dependent, three-dimensional models of mantle convection. *Geophys. Res. Lett.*, 20:2187–2190, 1993.

P. J. Tackley. Self-consistent generation of tectonic plates in time-dependent, three-dimensional mantle convection simulations. Part 1. Pseudoplastic yielding. *Geochem. Geophys. Geosys.*, 1:2000GC000036, 2000a. doi: 10.1029/2000GC000036.

P. J. Tackley. Self-consistent generation of tectonic plates in time-dependent, three-dimensional mantle convection simulations. Part2. Strain weakening and asthenosphere. *Geochem. Geophys. Geosys.*, 1: 2000GC000043, 2000b. doi: 10.1029/2000GC000043.

P. J. Tackley. Mantle geochemical geodynamics. In D. Bercovici, editor, *Treatise on Geophysics, Vol. 7: Mantle Dynamics*, pages 437–505. Elsevier, 2007.

P. J. Tackley. Modelling compressible mantle convection with large viscosity contrasts in a three-dimensional spherical shell using the yin-yang grid. *Phys. Earth Planet. Int.*, 171:7–18, 2008.

U. Trottenberg, C. W. Oosterlee, and A. Schüller. *Multigrid*. Academic Press, San Diego, 2001.

M. ur Rehman. *Fast iterative methods for the incompressible Navier-Stokes equations*. PhD thesis, Technische Universiteit Delft, 2009.

H. Uzawa. Iterative methods for concave programming. In K. Arrow and H. Uzawa, editors, *Studies in Linear and Non-linear Programming*, pages 154–165. Stanford University Press, 1958.

R. D. van der Hilst, S. Widiyantoro, and E. R. Engdahl. Evidence for deep mantle circulation from global tomography. *Nature*, 386:578–584, 1997.

R. Verfürth. A combined conjugate gradient-multigrid algorithm for the numerical solution of the Stokes problem. *IMA Journal of Numerical Analysis*, 4(4):441–455, 1984.

F. J. Vine and D. H. Matthews. Magnetic anomalies over oceanic ridges. *Nature*, 199:947–949, 1963.

U. Walzer and R. Hendel. Mantle convection and evolution with growing continents. *J. Geophys. Res.*, 113:B09405, 2008. doi: 10.1029/2007JB005459.

U. Walzer and R. Hendel. Predictability of Rayleigh-number and continental-growth evolution of a dynamic model of the Earth's mantle. In S. Wagner, A. B. M. Steinmetz, and M.Brehm, editors, *High Perf. Comp. Sci. Engng. Garching/Munich '07*, pages 585–600. Springer, Berlin, 2009.

U. Walzer and R. Hendel. A geodynamic model of the evolution of the Earth's chemical mantle reservoirs. In W. E. Nagel, D. B. Kröner, and M. M. Resch, editors, *High Perf. Comp. Sci. Engng. Stuttgart '10*, pages 573–592. Springer, Berlin, 2011.

U. Walzer, R. Hendel, and J. Baumgardner. Viscosity stratification and a 3D compressible spherical shell model of mantle evolution. In E. Krause, W. Jäger, and M. Resch, editors, *High Perf. Comp. Sci. Engng. '03*, pages 27–67. Springer, Berlin, 2004a.

U. Walzer, R. Hendel, and J. Baumgardner. The effects of a variation of the radial viscosity profile on mantle evolution. *Tectonophysics*, 384: 55–90, 2004b.

A. J. Wathen. Realistic eigenvalue bounds for the Galerkin mass matrix. *IMA Journal of Numerical Analysis*, 7:449, 1987.

A. Wegener. *Die Entstehung der Kontinente und Ozeane*. Vieweg, Braunschweig, 1915.

W.-S. Yang. *Variable viscosity thermal convection at infinite Prandtl number in a thick spherical shell*. PhD thesis, University of Illinois, Urbana-Champaign, 1997.

W.-S. Yang and J. R. Baumgardner. A matrix-dependent transfer multigrid method for strongly variable viscosity infinite Prandtl number thermal convection. *Geophys. Astrophys. Fluid Dyn.*, 92:151–195, 2000.

M. Yoshida. Possible effects of lateral viscosity variations induced by plate-tectonic mechanism on geoid inferred from numerical models of mantle convection. *Phys. Earth Planet. Int.*, 147:67–85, 2004.

M. Yoshida and T. Nakakuki. Effects on the long-wavelength geoid anomaly of lateral viscosity variations caused by stiff subducting slabs, weak plate margins and lower mantle rheology. *Phys. Earth Planet. Int.*, 172:278–288, 2009.

S. Zhong, M. Zuber, L. Moresi, and M. Gurnis. Role of temperature-dependent viscosity and surface plates in spherical shell models of mantle convection. *J. Geophys. Res.*, 105:11063–11082, 2000.

S. Zhong, A. McNamara, E. Tan, L. Moresi, and M. Gurnis. A benchmark study on mantle convection in a 3-d spherical shell using citcoms. *Geochem. Geophys. Geosys.*, 9:Q10017, 2008.

S. J. Zhong, D. A. Yuen, and L. N. Moresi. Numerical methods in mantle convection. In D. Bercovici, editor, *Treatise on Geophysics, Vol. 7: Mantle Dynamics*, pages 227–252. Elsevier, 2007.

Selbständigkeitserklärung

Ich erkläre, dass ich die vorliegende Arbeit selbständig und unter Verwendung der angegebenen Hilfsmittel, persönlichen Mitteilungen und Quellen angefertigt habe.

Jena, 10.01.2011